

SIMULATIONSTECHNIK UND -MODELLE IN DEN
SOZIAL- UND WIRTSCHAFTSWISSENSCHAFTEN

von

Christoph MANDL

Forschungsbericht Nr.110

Oktober 1976

Inhaltsverzeichnis

	Seite
Vorwort	1
1 KLASSIFIZIERUNG SOZIAL- UND WIRTSCHAFTSWISSENSCHAFTLICHER MODELLE (SWM)	3
1.1. Einleitung	3
1.2. Modellsprachen: Verbal-Simulation-Mathematik	3
1.3. Modellfunktionen: Zustandsanalyse-Prognose-Optimierung	7
1.4. Modellkomplexität und Anwendung der Modellarten in den einzelnen Fachbereichen	8
2 KLASSIFIZIERUNG VON SIMULATIONSMODELLEN	10
2.1. Einleitung	10
2.2. Deterministisch-stochastisch	10
2.3. Kontinuierlich-diskret	11
2.4. Stationär-instationär	12
2.5. Literatur	13
3 VON DER WIRKLICHKEIT ZUM SIMULATIONSMODELL, EIN BEISPIEL: BLACK JACK	14
3.1. Einleitung	14
3.2. Spielregeln und Problemstellung	14
3.3. Modellierung und Flußdiagramm	17
3.4. Problem I: Zufallszahlen	19
3.5. Das Computerprogramm	35
3.6. Problem II: Wie oft simulieren?	37
3.7. Literatur und Übungsaufgaben	39
4 OPTIMIERUNG	43
4.1. Einleitung	43
4.2. Spezielle Suchalgorithmen	46
4.3. Optimierung bei Black Jack	52
4.4. Literatur	54
4.5. Anhang: Das Computerprogramm	55

	Seite
5 METHODIK BEIM ERSTELLEN EINES SIMULATIONSMODELLS	64
5.1. Einleitung	64
5.2. Problemdefinition	65
5.3. Datenerfordernisse- und beschaffung	66
5.4. Formulierung der Subsysteme des Modells	68
5.5. Erstellung des Simulationsmodells aus den Teilsystemen	70
5.6. Schätzen der Simulationsvariablen- und parametern	70
5.7. Austesten des Modells	70
5.8. Validierung des Modells	71
5.9. Optimierung und Analyse der Resultate, Sensitivitätsanalyse	71
6 AMBULANZSYSTEM VON ZÜRICH	73
6.1. Einleitung	73
6.2. Notfallsysteme	73
6.3. Rettungssystem Zürich	77
6.4. Datenanalyse	78
6.5. Literatur	81
6.6. Aufgaben	81
7 SIMULATIONSSPRACHEN	82
7.1. Einleitung	82
7.2. Simulationssprachen und -modelle	82
7.3. Ein Beispiel einer Simulationssprache für diskrete Simulationen: SIMSCRIPT	85
7.4. Vergleich verschiedener Simulationssprachen	91
7.5. Literatur	95
8 ANALYSE VON SIMULATIONSRUNS	96
8.1. Einleitung	96
8.2. Definition	96
8.3. Methoden zur Ausschaltung unerwünschter transienter Zustände	97
8.4. Die Verwendung von Simulationsmodellen zur Analyse transienter Phänomene	100
8.5. Methode zur Analyse stochastischer Simulationsruns	101

III

	Seite
8.6. Anwendung auf Ambulanzsystem	110
8.7. Literatur	113
8.8. Anhang: Computerprogramme	114
 9 ÜBERGANG ZU KONTINUIERLICHEN MODELLEN	 130
9.1. Arten des Zeitfortschrittes	130
9.2. Wahl der Zeiteinheit	132
9.3. Dynamo	133
9.4. Literatur	135
 10 EIN MAKROÖKONOMISCHES MODELL ZUR SIMULATION VON KONJUNKTURZYKLEN	 136
10.1. Einleitung	136
10.2. Das ökonomische Modell	136
10.3. Das DYNAMO-Programm	147
10.4. Resultate und Kritik	156
10.5. Literatur	161
10.6. Aufgaben	161
 11 PERSONALPROGNOSEMODELL IN EINEM KRANKENHAUS	 162
11.1. Einleitung	162
11.2. Beschreibung des Ist-Zustandes	162
11.3. Das Simulationsmodell	166
11.4. Literatur	173

Vorwort

Dieses Skriptum entstand aus einer Vorlesung, welche ich im Jahre 1975/76 am Institut für Höhere Studien, Wien abhielt. Erste Ideen zum Vorlesungsaufbau erhielt ich noch während meiner Tätigkeit am Institut für Operations Research der Eidgenössischen Technischen Hochschule Zürich von Dr.T.Liebling, welchem ich an dieser Stelle hiefür danken möchte.

Simulation ist eine Technik, die, um sie mit Erfolg anwenden zu können, Erfahrung erfordert. Es ist daher wichtig, den Stoff nicht nur "im Prinzip" zu beherrschen, sondern sich der Mühe zu unterziehen, den Prozeß der Umsetzung von der analysierten Wirklichkeit bis zum funktionierenden Computerprogramm selbst durchzumachen. Eingedenk dieser Erkenntnis und, weil das Skriptum sich an Sozial- und Wirtschaftswissenschaftler genauso wie an Mathematiker richtet, erstere jedoch methodische, d.h. mathematische bzw. computernahe Probleme i.a. nur dann aufzunehmen bereit sind, falls diese Probleme im Zusammenhang mit der Analyse sozial- und wirtschaftswissenschaftlicher Modelle auftreten, wird das Skriptum an Hand von Fallstudien aufgebaut. Hierbei geht es aber auch darum, keine "Schulbeispiele" zu erörtern, sondern Probleme mit praktischer Relevanz, die tatsächlich primär durch Simulationsmodelle analysiert werden können. Vereinfachungen der Modelle, welche deren Wirklichkeitstreue in Frage stellen, sind jedoch wegen der im Rahmen einer Vorlesung beschränkt zur Verfügung stehenden Zeit kaum vermeidbar.

Um den klassischen betriebswirtschaftlichen Anwendungsbereich der Simulation, worüber bereits genügend Literatur existiert, zu verlassen, wird ein makroökonomisches und ein mit demographischen Modellen verwandtes Simulationsmodell behandelt werden.

An Grundvoraussetzungen zum Studium des Skriptums sind Kenntnisse in Statistik und FORTRAN erforderlich. Wenn es mir gelungen ist, mit diesem Skriptum dem Leser nicht nur die Methode zu vermitteln,

sondern auch klar zu machen, daß Simulationsmodelle eine wesentliche zusätzliche Hilfe bei der quantitativen Analyse sozialer bzw. ökonomischer Strukturen sein können, sofern man auch die methodischen und computertechnischen Probleme nicht übersieht, daß aber Simulation andererseits kein neues "Wundermittel" ist und von inhaltlichen Vorstellungen sehr wesentlich im Resultat geprägt wird, weswegen einige Skepsis bei Modellen im Stil des Weltmodells von J. Forrester angebracht erscheint, dann ist der Zweck dieses Skriptums voll erreicht.

1. KLASSIFIZIERUNG SOZIAL- UND WIRTSCHAFTSWISSENSCHAFTLICHER MODELLE (SWM)

1.1. Einleitung

Mit dem Werkzeug der Simulation kommt neben umgangssprachlichen sowie mathematisch-analytischen Modellen eine dritte Möglichkeit zur Beschreibung und Analyse unserer Gesellschaft ins Spiel. Es scheint daher angebracht, diese drei Modellarten etwas ausführlicher miteinander zu vergleichen, um somit bei einem gegebenen Problem besser abschätzen zu können, welche Art von Modell man dennam besten "basteln" soll. Die Auswahl der Modellart ist aber auch abhängig von der Antwort, welche das Modell liefern soll. Soll ein gegebener Zustand analysiert werden, soll eine Prognose zur Entwicklung gesellschaftlicher Prozesse erfolgen oder soll gar bei Entscheidungsspielraum eine optimale Handlungsweise gesucht werden?

Sicherlich kann kein klares Schema zur Modellklassifizierung sowie Modellauswahl gegeben werden, aber einige Kriterien sollen in den folgenden Kapiteln herausgearbeitet werden.

1.2. Modellsprachen

Komplexe Wirklichkeit läßt sich nur mittels Modelle beschreiben. Modelle entstehen dank der Fähigkeit des Menschen, aus einer Fülle von Eindrücken die, für das Geschehen wesentlichen Faktoren herauszukristallisieren, unwichtige Details zu vernachlässigen, Dinge mit gleichen Eigenschaften unter einem Begriff zu subsummieren, kurzum soziale und wirtschaftliche Realität in Worte zu fassen und damit zu beschreiben. Alle Modelle werden vom Menschen zunächst in dem ihm vertrautesten Medium, der Sprache, dargestellt. Sprachliche

Modelle haben den Vorzug, daß die gesamte, dem Menschen zugängliche Realität durch sie beschrieben werden kann - sie haben oftmals den Nachteil, daß nur sehr schwer, manchmal überhaupt nicht geprüft werden kann, ob diese Modelle auch annähernd richtig die Wirklichkeit wiedergeben, denn sprachliche Modelle sind unscharf, vieldeutig. Nur zu oft spielen beim Aufstellen sprachlicher Modelle emotionelle Beweggründe mit, so wenn es z.B. gilt, ein Modell zur Begründung des Antisemitismus aufzustellen, um dann, an Hand dieses Modells aufgestaute Agressionen auf einen hypothetischen Feind, in diesem Falle die Juden, abladen zu können. Unser Optimismus, scheint's, ist jedoch ungebrochen, denn in jeder Wissenschaft, mit der wir uns beschäftigen, nehmen wir ja implizit stets an, daß das menschliche Gehirn imstande ist, wirklichkeitsadäquate Modelle aufzustellen.

Die Überprüfung der Richtigkeit von sprachlichen Modellen ist, so sagte ich schon, schwierig. Zum Unterschied zu den Naturwissenschaften scheint es bei den Gesellschaftswissenschaften auch keinen allgemeinen Konsens über die Überprüfungsmethodik zu geben und so werden auch Modelle aufgestellt, die den Beweis ihrer Richtigkeit schuldig bleiben. Die naturwissenschaftliche Methodik besteht darin, aus der Analyse des erstellten Modells Thesen und Prognosen abzuleiten und diese dann mit der Realität zu vergleichen. So hat Einstein aus verschiedenen Experimenten heraus die spezielle Relativitätstheorie aufgestellt, daraus Behauptungen abgeleitet, die wiederum experimentell nachgewiesen wurden. Die Naturwissenschaften haben gegenüber den Sozial- und Wirtschaftswissenschaften einen großen Vorteil: Die aufgestellten Modelle beeinflussen nicht die Wirklichkeit. Die Anziehung von Massen wird nicht durch das Gravitationsgesetz beeinflußt. Aber welchen Einfluß haben

Konjunkturprognosen auf die Konjunktur? Diese Rückkoppelung SWM auf die Gesellschaft macht deren Überprüfung mitunter unmöglich. Dies sollte jedoch keineswegs die Bedeutung von Prognosen zum frühzeitigen Erkennen von Fehlentwicklungen mindern, selbst wenn diese Prognosen durch Gegenmaßnahmen nicht eintreten.

Nicht immer ist die Sprache das geeignete Medium zur Darstellung eines Modells und so hat der Mensch die Mathematik erfunden. Ursprünglich diente sie vor allem zur Konstruktion von Modellen der Sternenbewegung und zur Untersuchung geometrischer Figuren. Als adäquate Modellsprache hat sie dann primär die Physik, später auch die anderen Naturwissenschaften verwendet. In den Sozial- und Wirtschaftswissenschaften hat vor allem die Statistik Eingang gefunden und erst seit dem zweiten Weltkrieg gelangen andere Bereiche der Mathematik, vor allem in der Betriebswirtschaft und der Ökonomie, zur Anwendung. Und doch sind, mit Ausnahme der Statistik, der Anwendung der Mathematik in den Sozial- und Wirtschaftswissenschaften enge Grenzen gesetzt. Die Exaktheit und Eindeutigkeit der mathematischen Sprache verhindert oftmals das Umsetzen sprachlicher Modelle in mathematische. Und gelingt es doch, so sind die resultierenden Modelle so komplex, daß sie sich mit dem derzeitigen mathematischen Rüstzeug nicht analysieren lassen. Erfolg ist vor allem Modellen in Form linearer Gleichungs- oder Ungleichungssysteme beschieden, wie es Input-Output Modelle in der Ökonomie sowie lineare Optimierungsmodelle in der Betriebswirtschaft darstellen. Ausschlaggebend für diesen Erfolg ist nicht zuletzt die Entwicklung der Computertechnik, welche es ermöglicht, solche Modelle mit mehreren tausend Gleichungen bzw. Ungleichungen und mehreren zehntausend Variablen zu lösen. Nur leider passiert es auch, daß die sprachlichen Modelle, die diesen mathematischen zu Grunde liegen, so einfach sind, daß die mathematischen Modelle eine Scheingenauigkeit liefern, die in keinem Verhältnis zum unscharfen sprachlichen Modell steht. Mathematische Modelle haben den großen Vorteil, daß, sind sie einmal

formuliert und ist die Analysemethodik bekannt, allgemeingültige Aussagen und Schlüsse aus dem Modell gezogen werden können.

Zwischen den sprachlichen Modellen und den mathematischen klafft eine große Lücke. Viele sprachliche Modelle könnten exakter formuliert werden, einfacher analysiert, sind aber doch zu komplex, um sie als mathematisch-analytisches Modell darzustellen. Erst mit der Entwicklung von Computern konnte eine Lücke geschlossen werden in Form der Simulation. Simulation könnte man definieren als die Formulierung eines Modells in einer Computersprache ohne Umweg über ein mathematisches Modell. Wiewohl Computersprachen ein breiteres Anwendungsspektrum als Mathematik haben, sind es doch exakte Sprachen, in denen alle Größen eindeutig definiert sind. Dieser Bereich der unmittelbaren Umsetzung von Realität auf den Computer hat in den letzten 20 Jahren einen enormen Aufschwung erfahren, vor allem in der Betriebswirtschaft, aber auch im naturwissenschaftlich-technischen Bereich. Die Entwicklung eines Simulationsmodells ist i.a. einfacher als die Entwicklung eines mathematischen Modells für dasselbe Problem. Allgemeingültige Aussagen mittels Simulationsmodelle sind indes nicht möglich und die konkrete Analyse erfordert meist auch mehr Computerzeit als es für ein mathematisches Modell der Fall wäre. Deswegen gilt der Grundsatz, daß man versuchen soll, ein mathematisches Modell zu formulieren, um erst, wenn dies nicht gelingt, auf die Simulation zurückzugreifen. Simulation in den Gesellschaftswissenschaften kommt dem Experiment in den Naturwissenschaften in seiner Bedeutung recht nahe. Man kann gesellschaftliche Prozesse im Computer abbilden und dann damit experimentieren, also Varianten durchspielen.

Das Aufstellen von Simulationsmodellen setzt die Kenntnis einer allgemeinen Programmiersprache wie ALGOL, FORTRAN oder PL/I voraus. Hingegen sind zu deren Handhabung außer statistischem Wissen keine umfangreichen Mathematikkenntnisse vonnöten, wodurch sich ja Sozial- und Wirtschaftswissenschaftler oftmals von der Verwendung mathematischer Modelle abhalten lassen. Um das Übersetzen sprachlicher Modelle in eine Computersprache zu erleichtern, sind eine Unzahl von speziellen Simulationssprachen entwickelt worden, aber darauf werden wir später noch genauer eingehen. Wahrscheinlich ist trotz der vielen Worte der Begriff Simulation nicht viel klarer geworden und zu einem besseren Verständnis muß ich einstweilen auf das Kapitel 3, in dem wir ein einfaches Simulationsmodell erstellen werden, vertrösten.

1.3. Modellfunktionen

SWM können für unterschiedliche Fragestellungen konzipiert werden. Grundsätzlich lassen sich drei Kategorien unterscheiden: Zustandsanalysen, Prognosen und Optimierung. Diese drei Kategorien sind nicht trennbar. Um eine Prognose zu erstellen, muß man zumindest den derzeitigen Zustand eines Systems kennen. Um optimale Entscheidungen zu finden, muß man die Folgen alternativer Entscheidungen prognostizieren. Somit bedingt jede Modellkonstruktion zuerst eine Zustandsanalyse und in vielen Fällen läßt sich mangels Einsicht in die Dynamik gesellschaftlicher Prozesse kein Prognosemodell aufbauen. Aber gerade im Bereich von Zustandsanalysen können Simulations- oder auch mathematische Modelle weniger Hilfe bringen. Vor allem statistische Methoden werden für Zustandsanalysen herangezogen. Die eigentliche Domäne von Simulationsmodellen sind Prognosen und nur durch mathematische Modelle können gezielte Optimierungen vorgenommen werden. Die Optimierungsmethode mittels Simulationsmodelle bleibt das mehr oder weniger planmäßige Probieren.

1.4. Modellkomplexität

Die Liste der Modellkategorien wäre nicht vollständig, würde man nicht auch die Modelle nach Fachbereich aufgliedern. Sozial- und Wirtschaftswissenschaften sind ja untergliedert in Betriebswirtschaft, Ökonomie, Soziologie und Politologie und die Anwendung der oben charakterisierten Modelltypen in den einzelnen Bereichen ist sehr unterschiedlich. Generell scheint zu gelten, daß nichtsprachliche Modelle um so eher geeignet sind, um so mehr von individuellem menschlichem Verhalten abstrahiert werden kann und je kleiner der für das Modell relevante Bereich menschlichen Verhaltens ist. Hingegen scheint es bei der Anwendung nichtsprachlicher Modelle gleichgültig zu sein, ob viele oder wenige Menschen an dem untersuchten gesellschaftlichen Prozeß beteiligt sind. Wohl existieren mathematische Modelle zur Beschreibung wirtschaftlichen Verhaltens von Menschengruppen, hingegen werden gruppendynamische Prozesse mit nur wenigen beteiligten Personen, bei denen aber sehr viele Verhaltensweisen der Beteiligten ins Kalkül gezogen werden, eher durch sprachliche Modelle beschrieben. Dementsprechend haben mathematische und Simulationsmodelle vor allem in der Betriebswirtschaft und der Ökonomie Eingang gefunden, wohingegen soziologische und politologische nichtsprachliche Modelle vergleichsweise spärlich sind, klammert man militärstrategische Modelle aus und rechnet man Weltmodelle im Stil von Forrester oder Mesarovich mehr zur Ökonomie, als zur Politologie. Ein soziologischer Bereich in dem sich Simulation als sinnvoll erwiesen hat, sind demographische Prognosemodelle. Wir werden im Rahmen der Vorlesung auch ein solches studieren.

Die Anwendung von Modellen nach den Kategorien Zustandsanalyse-Prognose-Optimierung scheint sich hingegen nicht auf bestimmte Fachbereiche zu beschränken. Gleichwohl sind optimale Entscheidungen im betriebswirtschaftlichen Bereich sicher einfacher zu finden, als z.B. im politischen Bereich.

Die Darlegungen dieses Kapitels kurz zusammenzufassen dient die folgende Tabelle über die Anwendung der beschriebenen Modellkategorien:

Betriebs- wirtschaft	Sprache	Zustand Prognose Optimierung	oft oft oft
	Simulation	Zustand Prognose Optimierung	nicht oft oft
	Mathematik	Zustand Prognose Optimierung	stat.Methoden oft oft
Ökonomie	Sprache	Zustand Prognose Optimierung	oft oft oft
	Simulation	Zustand Prognose Optimierung	nicht selten ?
	Mathematik	Zustand Prognose Optimierung	stat.Methoden oft ?
Soziologie	Sprache	Zustand Prognose Optimierung	oft oft oft
	Simulation	Zustand Prognose Optimierung	nicht sehr selten nicht
	Mathematik	Zustand Prognose Optimierung	stat.Methoden selten nicht
Politologie	Sprache	Zustand Prognose Optimierung	oft oft oft
	Simulation	Zustand Prognose Optimierung	nicht ? nicht
	Mathematik	Zustand Prognose Optimierung	stat.Methoden ? nicht

2.

KLASSIFIZIERUNG VON SIMULATIONSMODELLEN

2.1. Einleitung

Die grobe Einteilung SWM im allgemeinen sollte helfen, die Stellung von Simulationsmodellen im Gesamtrahmen besser zu begreifen. Um die Aufgliederung der Vorlesung besser zu verstehen, ist es auch notwendig, die Simulationsmodelle nach deren Charakteristiken zu unterteilen. Nunmehr kommen bei der Einteilung auch programmiertechnische Unterschiede zum Tragen, denn langsam aber sicher begeben wir uns in den Bereich der Simulation und damit auch des Computers.

2.2. Deterministische-Stochastische Modelle

Schwankungen der Werte von Kenngrößen des untersuchten Modells können oftmals nicht durch das Modell selbst erklärt werden. Solche Wertveränderungen bezeichnen wir dann als zufällig und die Größe stochastisch. Vor allem in mathematischen Modellen werden Zufallsgrößen dann durch Verteilungsfunktion, Erwartungswert oder Varianz charakterisiert und mit diesen nunmehr nicht zufälligen, also deterministischen Größen operiert. In Simulationsmodellen ist es oft nicht ausreichend, mit deterministischen Größen zu arbeiten. Es müssen auch die Zufälligkeiten simuliert werden. Werden also in der Simulation Größen verwendet, die zufällige Werte erhalten, so nennt man dies eine stochastische Simulation. Stochastische Simulationen sind ungleich aufwendiger als deterministische, da man, um brauchbare Resultate zu erhalten, dieselbe Simulation öfter durchführen muß, weil zunächst die Ergebnisse stochastischer Simulation wiederum zufällig sind und erst durch die Wiederholung Gesetzmäßigkeiten in Form von Verteilungsfunktionen oder Erwartungswert geschätzt werden können. Der Entscheid, ob man für eine gestellte Aufgabe eine deterministische oder stochastische Simulation vornehmen soll, ist schwierig, da abzuschätzen ist, inwieweit

das Arbeiten mit deterministischen Größen wie Erwartungswert und Varianz, die Realitätsnähe des Modells verfälscht. Wir werden im nächsten Kapitel sehen, daß zur Simulation von Black Jack ein deterministisches Simulationsmodell sinnlos wäre.

2.3. Kontinuierliche - diskrete Modelle

Ein wichtiger Begriff bei der Formulierung eines Simulationsmodells ist das Ereignis. Unter Ereignissen versteht man Veränderungen des Zustands des Simulationsmodells zu einem bestimmten Zeitpunkt. Das heißt nicht, daß in der Realität, der das Simulationsmodell entspricht, zwischen zwei Ereignissen Nichts passiert, vielmehr sind im Modell gewisse Vorgänge zu einem Zustand zusammengefaßt. Ist in einem Simulationsmodell des Verkehrsverhaltens auf einer Autobahn als Zustand das Verbleiben eines Autos in seiner Fahrspur definiert, so wäre ein Ereignis im Sinne der Simulation das Wechseln der Fahrspur. Ereignisse können nun zu zufälligen, erst im Verlauf der Simulation sich herauskristallisierenden Zeitpunkten stattfinden, dann spricht man von diskreter Simulation. Die meisten Modelle, welche Prozesse im Detail studieren, insbesondere betriebswirtschaftliche Modelle, wie Lagerhaltung, Produktionsprozesse, Verkehrsmodelle, sind diskrete Simulationsmodelle. Sind die untersuchten gesellschaftlichen Prozesse komplexer, so läßt sich gar nicht mehr der genaue Zeitpunkt eines Ereignisses festlegen. Man vernachlässigt dann die Zufälligkeit der Zeitpunkte von Ereignissen und legt a priori für das Modell fest, daß sich der Modellzustand alle Vierteljahr oder alle Jahre verändern soll. Diese, die kontinuierlichen Simulationsmodelle, sind typisch für ökonomische Modelle darunter auch die "Weltmodelle", aber auch für demographische Modelle. Vom programmiertechnischen Standpunkt aus sind die

diskreten Simulationen, wiewohl sie einfachere Prozesse beschreiben, die komplizierteren, da bei diesen auch Zeit eine Variable ist und nicht, wie bei kontinuierlichen Simulationen von vorne herein festgelegt ist. - Natürlich gibt es auch Modelle, in denen die Zeit kein Modellparameter ist. Ein Beispiel wäre das im kommenden Kapitel besprochene Black Jack oder überhaupt alle Kartenspiele. Modelle dieser Art sind ihrer Struktur nach jedoch eher verwandt mit diskreten als mit kontinuierlichen Simulationsmodellen. Die Unterscheidung diskrete-kontinuierliche Simulation ist vor allem für die Wahl einer speziellen Simulationssprache wichtig, da man mit einer Simulationssprache, welche sich für kontinuierliche Simulationen eignet, keine diskrete Simulation durchführen kann, wohingegen das Umgekehrte nicht unbedingt gilt. Wiewohl programmiertechnisch schwieriger, werden wir in der Vorlesung mit diskreten Simulationsmodellen beginnen, da für diese wiederum das Umsetzen von Realität ins Modell ungleich einfacher ist, als bei komplexen ökonomischen oder demographischen Prozessen, für die wir dann anschließend kontinuierliche Simulationsmodelle konstruieren werden.

2.4. Stationäre - instationäre Modelle

Erstellt man ein stochastisches Simulationsmodell, so muß man bei der Auswertung darauf achten, ob es sich um ein stationäres oder ein instationäres Modell handelt. Ein stationäres Modell beinhaltet nur solche zufällige Größen, bei denen die Gesetzmäßigkeiten mit denen die zufälligen Werte auftreten, also z.B. die Verteilungsfunktionen, sich nicht mit der Zeit ändern, also stationär bleiben. Da instationäre stochastische Simulationsmodelle sehr unangenehme statistische Fragen aufwerfen, vermeidet man, wenn irgendmöglich, solche zu erstellen. Dabei nimmt man jedoch in Kauf, daß man mit stationären Modellen im allgemeinen nur kurzfristige Prognosen erstellen kann, da in der Realität die wenigsten Prozesse stationär sind,

wenngleich sich manche genügend langsam verändern, um sie durch stationäre Modelle zu approximieren. Bei einer Analyse des Ambulanzsystems von Zürich wurde z.B. die Notfallhäufigkeit in den einzelnen Bezirken als zufällige Größe, die einer Poissonverteilung gehorcht, ins Modell eingebracht. Zürich ist nun eine Stadt mit starker Expansion, was sich auch auf die Bevölkerungsverteilung der einzelnen Bezirke auswirkt, weswegen zu erwarten ist, daß die Notfallhäufigkeit sich mit der Zeit verändert. Unter Vernachlässigung dieser Einsicht zeigte sich, daß auch ein stationäres Modell die letzten drei Jahre hinreichend gut beschreiben kann.

2.5. Literatur

Emshoff, J.R. & Sisson, R.L., Simulation mit dem Computer, München, Verlag Moderne Industrie, 1972

Harbordt, St., Computersimulation in den Sozialwissenschaften, Band 1 & 2, rororo Studium, Nr.49-50, Hamburg, 1974

3. VON DER WIRKLICHKEIT ZUM SIMULATIONSMODELL, EIN BEISPIEL: BLACK JACK

3.1. Einleitung

Wir können nun daran gehen unser erstes Simulationsmodell aufzubauen. Hierbei werden wir auf einige Probleme stoßen, die wir dann auch gleich in aller Ausführlichkeit erörtern werden. Das gewählte Spiel Black Jack hat den Vorteil, daß die wirklichkeitsgetreue Abbildung auf ein Simulationsmodell keine allzugroßen Schwierigkeiten bietet. Trotzdem ist das Problem, das wir uns dabei stellen werden, alles andere als trivial und hat schon findige Mathematiker beschäftigt. Nach der Darlegung der Spielregeln sowie der Problemstellung werden wir daran gehen, ein FORTRAN-Programm zu erstellen. Hierbei müssen wir uns länger über das Erzeugen von Zufallszahlen am Computer unterhalten. Haben wir das Simulationsprogramm zum Laufen gebracht, werden wir auch darüber nachdenken müssen, wie lange wir denn simulieren sollen, um relevante Aussagen zu erhalten. Schließlich wird uns die Frage beschäftigen, wie wir vorgehen sollen, um mittels des Simulationsmodells optimale Lösungen unseres Problems zu finden. Vieles von dem, was wir dabei lernen werden, gilt allgemein für Simulationsmodelle, weswegen Black Jack, wiewohl nur ein Spiel (obwohl eines bei dem man Geld gewinnen, aber auch verlieren kann), ein gutes Lernbeispiel ist.

3.2. Spielregeln und Problemstellung

Zwei bis sieben Spieler spielen gegen den Croupier mit dem Ziel, Karten bis 21 oder so nahe wie möglich an 21 zu ziehen, ohne dabei den Gesamtwert von 21 zu überschreiten. Nach Tätigung der Einsätze gibt der Croupier die Karten offen aus - zuerst eine an jeden Spieler (im Uhrzeigersinn), zuletzt sich selbst eine. Danach erhält jeder Spieler - ebenfalls offen - eine

zweite Karte. Asse zählen 1 (eins) oder 11 (elf), Bilder zählen 10 (zehn), alle anderen Karten zählen ihren normalen Wert. Hat der Spieler Black Jack (ein As mit einem Bild oder einer Zehn), so ist dies die höchste Paarung, die 3:2 ausbezahlt wird und nur von einem Black Jack des Croupiers ausgeglichen werden kann. Dies bedeutet, daß der Satz weder verliert noch gewinnt. Kommt kein Black Jack zustande, versucht der Spieler, so nahe wie möglich an 21 heranzukommen, indem er mit der Erklärung "Karte" oder "nein" beliebig viele Karten vom Croupier fordert. Mit dem Ausdruck "nein" gibt er zu verstehen, daß er keine weiteren Karten wünscht.

Hat der Croupier 16 Punkte oder weniger, muß er ziehen, hat er 17 oder mehr, muß er bleiben.

Ist der Spieler näher an 21 als der Croupier, gewinnt er die Höhe seines Einsatzes. Ist der Gesamtwert seiner Karten niedriger als der des Croupiers, verliert er seinen Einsatz. Ist der Gesamtwert der Karten des Spielers identisch mit dem Wert der Karten des Croupiers, kann der Satz, der weder gewinnt noch verliert, stehen bleiben, zurückgezogen oder verändert werden. Überschreitet der Spieler während des Ziehens den Gesamtwert 21, verliert er.

Das Kartenpaket, mit dem gespielt wird, besteht aus gleich viel Zweiern, Dreiern, ... , Assen, insgesamt ca. 260 Karten. Die verwendeten Karten kommen nicht zum Kartenpaket dazu, erst wenn bei einem Spielbeginn nur mehr ca. 50 Karten übrig sind, werden alle Karten wieder zusammengemischt und dann mit dem gesamten Kartenpaket gespielt.-Soweit die Spielregeln, zu denen es noch zusätzliche Möglichkeiten für die Spieler gibt, die aber am Spielgeschehen nichts prinzipiell ändern, die wir daher der Einfachheit halber außer Betracht lassen.

Black Jack gilt als Glücksspiel und wird üblicherweise in den Spielkasinos gespielt. Es ist aber das einzige Glücksspiel, bei dem zunächst unklar ist, worin die Chance der Bank besteht, denn während der Bankier durch die Spielregeln in seinem Spielverhalten vollständig determiniert ist, hat der Spieler freie Hand zu entscheiden, ob er noch Karten will oder nicht. Uns, als potentielle Kasinobesucher interessiert nun:

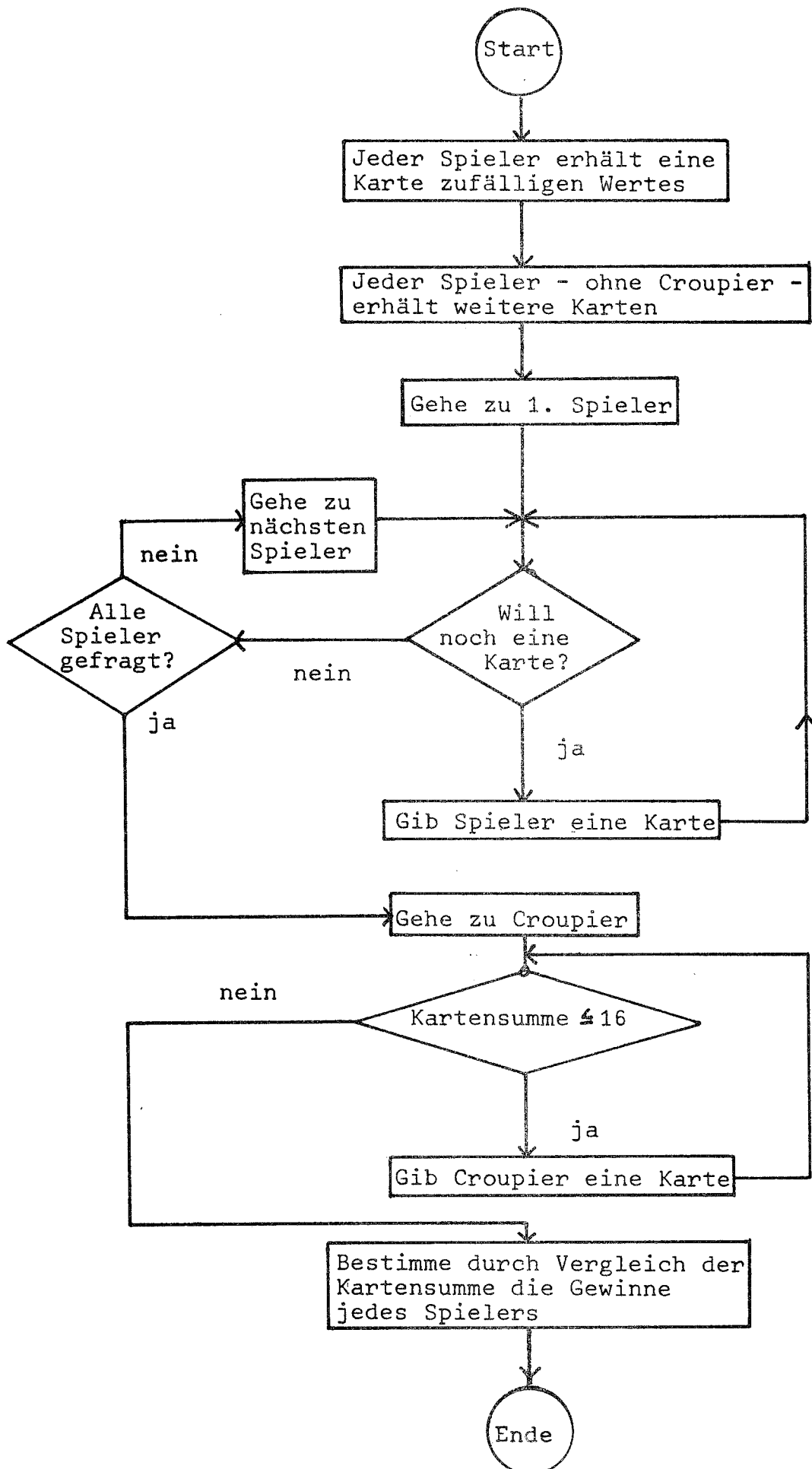
- 1) Wie groß ist die mittlere Verlustwahrscheinlichkeit jedes Spielers, falls er dieselbe Strategie, wie der Bankier wählt.
- 2) Gibt es ein optimales Spielverhalten, bei dem die Verlustwahrscheinlichkeit minimiert wird und wie sieht diese Optimalstrategie aus?

Vier Möglichkeiten bieten sich uns, Antwort auf die gestellten Fragen zu finden. Wir können uns zusammensetzen und spielen und verschiedene Strategien ausprobieren, bis wir meinen, die beste gefunden zu haben. Diese Möglichkeit ist sehr zeitraubend, denn jedes Spiel dauert einige Minuten und, um Strategien zu testen, Erwartungswerte zu schätzen, benötigen wir wohl einige tausend Spiele - ein etwas mühsames Unterfangen. Außerdem wollen wir ja den sozial- und wirtschaftswissenschaftlichen Ernstfall proben, bei dem es höchst selten möglich ist, verschiedene Verhaltensweisen in der Wirklichkeit auszuprobieren, schön der Reihe nach. Kein Politiker, kein Manager würde diesen Vorschlag akzeptieren. Wir können auch versuchen, mit einem umfangreichen mathematischen Apparat an das Problem heranzutreten, insbesondere denke ich hierbei an die Beschreibung des Spiels in Form eines Markoffprozesses mit wohldefinierten Zuständen und Übergangswahrscheinlichkeiten. Aber, gesetzt den Fall, das Problem ließe sich auf diese Art lösen, was gar nicht sicher ist, so wäre der Aufwand an Denkarbeit groß, denn ohne ein paar gute Einfälle ginge es kaum. Wir können weiters schlicht die Literatur studieren. Vielleicht hat sich schon jemand vor uns an die Lösung gewagt und ist ihm auch gelungen. Dieser Weg

ist oftmals der erfolgversprechenste, auch in unserem Fall, aber wir wollen ihn ignorieren. Schließlich können wir das Spiel dem Computer verständlich machen, also ein Programm schreiben, den Computer für uns spielen lassen, kurz, wir können simulieren und das wollen wir auch tun.

3.3. Modellierung und Flußdiagramm

Die Spielregeln sollen also nun in eine Programmiersprache (wir wählen hier FORTRAN) übersetzt werden. Bevor wir die einzelnen FORTRAN-Statements formulieren, wollen wir den logischen Ablauf in Form eines Flußdiagramms darstellen, dies erleichtert die eigentliche Programmierung und schützt vor logischen Fehlern.



Das grobe Flußdiagramm in Figur 3-1 bedarf noch detaillierter Flußdiagramme für einige Teilbereiche. Betrachten wir den ersten Schritt. Er besteht darin, einen zufälligen Kartenwert jedem Spieler zuzuweisen. Dieser Kartenwert ist eine Zahl zwischen 1 und 13, wenn wir As mit 1, Bub mit 11, Dame mit 12 und König mit 13 codieren. Wir müssen also nun ein Verfahren kennenlernen, wie wir zufällig ganze Zahlen zwischen 1 und 13 am Computer erzeugen können. Damit werden wir uns im nächsten Kapitel ausführlich auseinandersetzen.

3.4. Problem I: Zufallszahlen

Bevor wir Methoden zu Generierung von Zufallszahlen betrachten, ist es nützlich, sich die Eigenschaft, welche diese Zufallszahlen ausweisen müssen, klar zu machen. Wir sagen, ein Prozeß erzeuge Zufallszahlen, falls er n diskrete Zahlen erzeugt, wobei jede dieser Zahlen die gleiche Wahrscheinlichkeit, nämlich $1/n$, besitzt, erzeugt zu werden, und falls jede neue Zahl vollkommen unabhängig von früher erzeugten Zahlen ist. Statistisch heißt das, daß die Zahlen gleichverteilte, unabhängige Variablen sind.

Eine Anzahl Tests wurden entwickelt um zu testen, ob eine Folge von Zahlen dem Kriterium der Zufälligkeit genügt. Bekanntlich können beim Testen von Hypothesen diese nicht mit absoluter Sicherheit verworfen werden, falls der statistische Test negativ ausfällt, alles was man sagen kann ist, daß, falls die Hypothese doch wahr ist, die beobachteten

Resultate ein unwahrscheinliches Ereignis darstellen. Die Wahl der Verwerfungswahrscheinlichkeit definiert, wie unwahrscheinlich ein solches Ereignis sein muß, bevor die Hypothese verworfen wird (Fehler 1. Art). Natürlich gibt es Fälle, wo solch seltene Ereignisse eintreffen und je mehr Tests wir eine Folge von Zahlen unterwerfen, desto größer wird die Wahrscheinlichkeit, daß diese durch einen dieser Tests als nicht zufällig verworfen wird. Daher sollten nur die Eigenschaften von Zufallszahlen, welche kritisch sind für die Schlüsse, welche aus einer Simulation gezogen werden, getestet werden und nur eine sehr große Stichprobe solcher Zufallszahlen hiezu verwendet werden.

Angenommen, wir wollen testen, ob eine Folge von Zahlen R_1, R_2, \dots, R_m dem Kriterium der Zufälligkeit genügt. Wenn jede Zahl R_i eine Dezimalzahl mit n Ziffern ist, kann auf 2 Arten getestet werden:

- (1) Man testet jede Zahl in ihrer Gesamtheit (wobei es 10^n mögliche Zahlenwerte gibt)
- (2) Man testet jede Ziffer der Zahl R_i einzeln also $r_{i1}, r_{i2}, \dots, r_{in}$ (Hier ist die Zahlenfolge so, daß der höchsten Ziffer von R_{i+1} die niedrigste Ziffer von R_i folgt).

Die Vorteile der ersten Methode sind erstens, daß für eine gleiche Zahl von Beobachtungen Tests auf Gleichverteilung über eine längere Folge von Zahlen durchgeführt werden können und daß zweitens der Test sich implizit primär auf das Testen der höheren Ziffern in den Zahlen konzentriert. Der Nachteil dabei ist, daß Zahlen als zufällige akzeptiert werden können, die dieses Kriterium für die kleineren Ziffern absolut nicht erfüllen. Die zweite Methode wertet jede Ziffer als gleich wichtig und wird daher bevorzugt, falls die niedrigeren Ziffern in einer Simulation von Bedeutung sind.

Die am meisten verwendeten Tests sind der Häufigkeitstest und der Serien Test . Beide basieren auf der χ^2 -Statistik. Der Häufigkeitstest stellt fest, ob alle Zahlen mit gleicher Wahrscheinlichkeit erzeugt werden, er analysiert nicht die Unabhängigkeit der Zahlen voneinander. Als Beispiel der Anwendung dieses Tests nehmen wir an, wir haben eine Folge von M einziffrigen Zufallszahlen.

Sei f_i = Häufigkeit, mit der Ziffer i in der Folge auftritt
($i = 0, 1, \dots, 9$)

E_i = erwartete Häufigkeit mit der Ziffer i aufträte, wäre die Folge vollkommen zufällig.

Dann ist der Häufigkeitstest

$$\chi_F^2 = \sum_{i=0}^9 \frac{(f_i - E_i)^2}{E_i} \quad (3-1)$$

Klarerweise wäre $E_i = \frac{M}{10}$ bei einer Folge gleichverteilter Zufallszahlen. Die Statistik ist verteilt nach χ^2 mit, in diesem Fall, 9 Freiheitsgraden. Durch Nachschlagen der χ^2 -Tafel in einem Statistikhandbuch und mit der zu fixierenden Wahrscheinlichkeit für einen Fehler erster Art, den wir riskieren wollen, können wir feststellen, ob wir die Testhypothese verwerfen müssen oder nicht.

Der Serien Test dient zur Feststellung , ob Paare, Tripel etc. von Ziffern zufällig auftreten. Dies ist der primäre Test auf Unabhängigkeit einer Folge von Ziffern. Zur Illustration nehmen wir an, wir wollen die Unabhängigkeit von Ziffernpaaren in einer Folge von M einziffrigen Zufallszahlen testen.

Sei f_{ij} = Häufigkeit von Paaren in der die erste Ziffer i und die darauf folgende Ziffer j ist

E_{ij} = Erwartete Anzahl von Paaren (i,j), falls die Zufallsreihe vollständig zufällig und unabhängig wäre.

Dann ist der Serien Test

$$\chi^2_F = \sum_{i=0}^9 \sum_{j=0}^9 \frac{(f_{ij} - E_{ij})^2}{E_{ij}} - \frac{\left(\sum_{j=0}^9 f_{ij} - \sum_{j=0}^9 E_{ij} \right)^2}{\sum_{j=0}^9 E_{ij}} \quad (3-2)$$

wobei natürlich

$$E_{ij} = M/100 \quad \text{und} \quad \sum_{j=0}^9 E_{ij} = M/10 \quad .$$

(3-2) ist aber nun χ^2 -verteilt mit 90 Freiheitsgraden (Anzahl Klassen in den Zahlenpaaren minus der Anzahl Klassen in den einzelnen Zahlen, $100 - 10$). Analog zum Häufigkeitstest kann nun die Hypothese getestet werden.

Nach diesem kurzen Abriß zum Testen von Zufallszahlen, wollen wir kurz erläutern, wie Zufallszahlen erzeugt werden. Wir beschränken uns gleich auf das Erzeugen von Zufallszahlen auf digitalen Rechenanlagen. Daneben kann man durch physikalische Prozesse oder schlicht durch Würfeln Zufallsfolgen erstellen. Zufallszahlengeneratoren basierend auf mathematischen Beziehungen sind nicht echt zufällig, da die Folge vollständig deterministisch ist. Trotzdem bestehen diese Folgen die meisten statistischen Tests und haben dabei den bei Simulationsmodellen wichtigen Vorteil, daß die Folge reproduzierbar ist. Sobald nämlich in solch einer Zufallsfolge der Input für den Zufallszahlengenerator der gleiche, wie zu einem früheren Zeitpunkt ist, wird die Folge ab diesem Punkt identisch mit der früher erzeugten. Die Anzahl Zahlen, die bis zur Wiederholung einer Zufallsfolge erzeugt werden, nennt man die Zykluslänge eines Generators. Die so erzeugten Zahlen heißen Pseudozufallszahlen.

Die heute gebräuchlichste Methode zum Erzeugen einer ganzzahligen Zufallszahl heißt congruential method. Hat man eine Zufallszahl X_i , so erhält man die folgende Zufallszahl X_{i+1}

$$X_{i+1} = aX_i + c \pmod{m} \quad (3-3)$$

Hierbei sind a , c und m fest vorgegebene ganze Zahlen. $(\text{mod } m)$ bedeutet, daß die Zahl $a \cdot X_i + c$ durch m dividiert wird. Der dabei verbleibende ganzzahlige Rest ist dann X_{i+1} . Daher wird durch m die Größe der Zufallszahlen beschrieben, da diese stets im Intervall $[0, m-1]$ liegen, setzt man alle Zahlen in (3-3) als nicht negativ voraus. Üblicherweise nimmt man für m die jeweils auf einem Computer größte darstellbare ganzzahlige Zahl. a und c sind beliebig wählbar; durch diese Zahlen wird die Zykluslänge der Pseudozufallszahlenfolge festgelegt. Muß a stets positiv, also größer gleich 1 ($a \geq 1$) sein, so darf c auch Null sein ($c \geq 0$). Im Fall $c=0$ heißt (3-3) auch multiplikative Methode, für $c \neq 0$ gemischte Methode. Ein Vergleich der zwei Methoden erbrachte, daß

- die gemischte Methode eine größere Zykluslänge aufweist
- die multiplikative Methode mehr statistische Tests besteht
- die multiplikative Methode i.a. schneller ist, also weniger Computerzeit erfordert.

Üblicherweise besitzt ein Computer einen Zufallszahlengenerator in seiner Software und so brauchen wir uns nicht detaillierter mit der Bestimmung der Größen a, c und m auseinandersetzen. Wichtiger, da bei Simulationen häufig auftretend, ist das Problem, wie man gleichverteilte Zufallszahlen in solche mit einer beliebigen Verteilungsfunktion konvertieren kann.

Inverse Transformation

Betrachten wir eine beliebige, nicht notwendigerweise gleichverteilte Zufallsgröße X . Diese wird charakterisiert durch Angabe ihrer Verteilungsfunktion, die definiert ist durch

$$\begin{aligned} F(x) &= \text{Wahrscheinlichkeit, daß zufälliger Wert der} \\ &\quad \text{Zufallsgröße } X \text{ kleiner oder gleich } x \text{ ist, wo} \\ &\quad x \text{ beliebige aber feste Zahl aus } (-\infty, \infty) \\ &= P\{X \leq x\} \quad x \in (-\infty, \infty) \end{aligned} \quad (3-4)$$

Sei nun $F(x)$ eine stetige, streng monotone Funktion, dann gilt

$$\begin{aligned} x \leq y &\iff F(x) \leq F(y) \\ 0 &\leq F(x) \leq 1 \end{aligned} \quad (3-5)$$

Ist Z eine in $[0,1]$ gleichverteilte Zufallszahl, so gilt für die zugehörige Verteilungsfunktion $U(x)$

$$\begin{aligned} U(x) &= 0 \quad x < 0 \\ U(x) &= P\{Z \leq x\} = x \quad 0 \leq x \leq 1 \\ &\Rightarrow U(x) = P\{Z \leq U(x)\} \\ U(x) &= 1 \quad x \geq 1 \end{aligned} \quad (3-6)$$

Wir wollen uns nun fragen nach welcher Verteilungsfunktion $F(X)$ verteilt ist, falls X nach $F(x)$ verteilt ist und $F(x)$ streng monoton und stetig ist.

Es gilt ja

$$F(x) = P\{X \leq x\}.$$

Wegen (3-5) gilt daher

$$F(x) = P\{X \leq x\} = P\{F(X) \leq F(x)\}. \quad (3-7)$$

Wegen (3-6) folgt aber aus Beziehung (3-7), daß $F(X)$ gleichverteilt ist. Also gilt für eine strenge monotone und stetige Verteilungsfunktion $F(x)$, daß

$$Y = F(X)$$

Y gleichverteilt nach (3-6)

(3-8)

X verteilt nach F(x)

Hat man daher einen Zufallszahlengenerator, welcher gleichverteilte Zufallszahlen Y erzeugt, so läßt sich daraus eine beliebige verteilte Zufallszahl X mittels

$$X = F^{-1}(Y)$$

(3-9)

darstellen.

Beispiel:

Die Verteilungsfunktion der Exponentialverteilung (mit $\lambda=1$) lautet

$$F(x) = 1 - e^{-x} \quad x \geq 0$$

Mit (3-5) und (3-6) erhält man daher

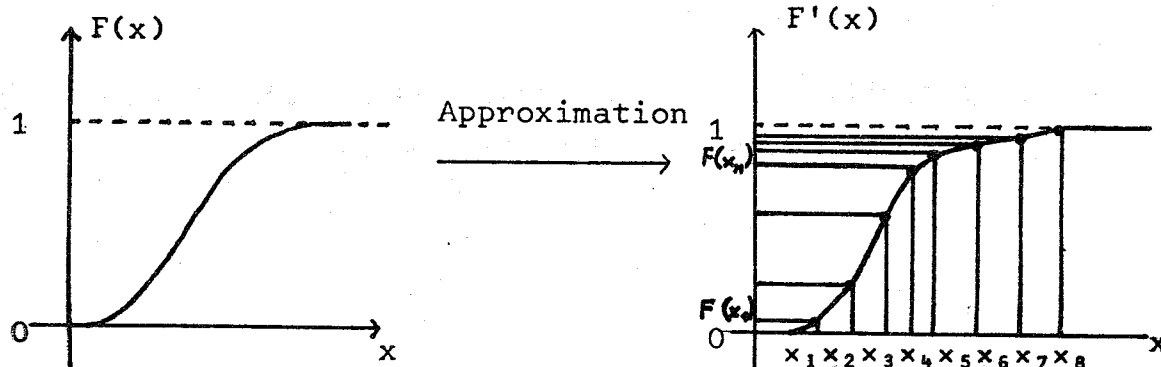
$$r = F(x) = 1 - e^{-x}$$

$$x = -\ln(1-r).$$

Sofern möglich, sollte man stets obige Methode verwenden, da sie am effizientesten arbeiten. Bei einigen Verteilungsfunktionen ist aber deren Inverse nicht angebar. In diesem Falle kann man sich mit der linearen Approximationsmethode behelfen.

Lineare Approximationsmethode

Hat man eine streng monotone, stetige Verteilungsfunktion $F(x)$ gegeben, deren Inverse nicht explizit angebar ist, so approximiert man diese abschnittsweise durch Gerade, also



In einem bestimmten Abschnitt, sagen wir

$$x_n \leq X \leq x_{n+1},$$

lautet dann die Gleichung der Verteilungsfunktion

$$F(x) - F'(x) = F(x_n) + \frac{F(x_{n+1}) - F(x_n)}{x_{n+1} - x_n} \cdot (x - x_n) \quad (3-10)$$

Diese Verteilungsfunktion ist aber explizit invertierbar und daraus lässt sich, falls gilt

$$F(x_n) \leq r \leq F(x_{n+1}) \quad \text{und} \\ r \text{ verteilt nach } U(r), \quad (3-11)$$

X berechnen durch

$$X = x_n + (x_{n+1} - x_n) \frac{r - F(x_n)}{F(x_{n+1}) - F(x_n)} \quad (3-12)$$

Man erzeugt also wiederum eine gleichverteilte Zufallszahl r, sucht jenen Bereich der linearen Approximation für den (3-11) gilt und berechnet dann X mittels (3-12).

Methode der verallgemeinerten Umkehrfunktion

Sei nun F(x) eine beliebige Verteilungsfunktion. Definiere

$$\underline{g}(z) = \inf\{u | F(u) \geq z\}$$

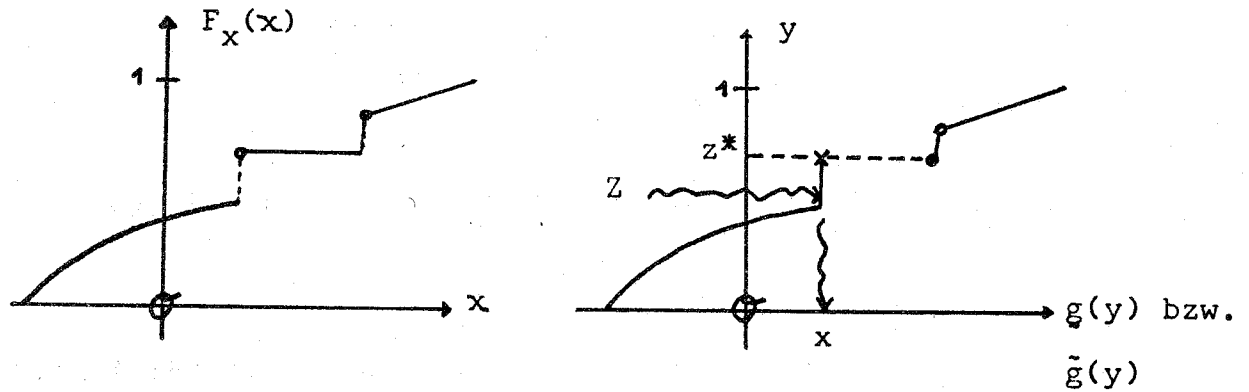
bzw. $\tilde{g}(z) = \sup\{u | F(u) \leq z\}$

Dann ist

$$X := g(Z)$$

mit $g = \underline{g}$ oder $= \tilde{g}$

eine nach F(x) verteilte Zufallsgröße, falls Z eine in [0,1) gleichmässig verteilte Zufallsgröße ist.



g und \tilde{g} unterscheiden sich nur im Punkte z^* : $g : x, \tilde{g} : \bullet$

Dank der Definition von $g(Z)$ gilt

$$P\{g(Z) \leq x\} = z = F(x)$$

wovon man sich durch geeignete Fallunterscheidungen überzeugt.
Es ergibt sich der

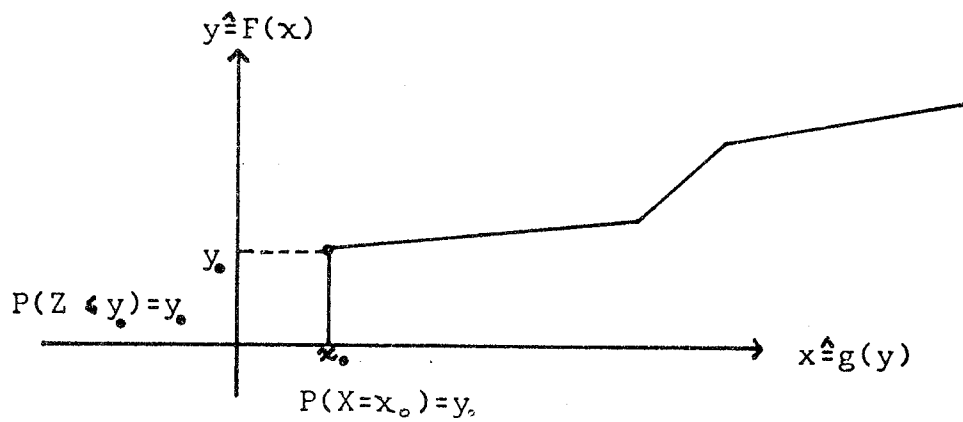
Algorithmus

(1) Würfle Z , in $[0,1)$ gleichverteilt

(2) $X := g(Z)$

ENDE

Folgende Zeichnung deutet einerseits den Verlauf des Algorithmus an, andererseits darauf, daß die X -Werte mit den richtigen Wahrscheinlichkeiten ausgewürfelt werden.



Bemerkung: ist $F(x)$ stetig und streng monoton, so fällt $g(\cdot)$ mit $F^{-1}(\cdot)$ zusammen.

Diskrete Zufallsgrößen

Sei X eine Zufallsgröße mit der Wahrscheinlichkeitsverteilung

$$P\{X = x_i\} = p_i \quad i=1, \dots, M$$

Das Verfahren der verallgemeinerten Umkehrfunktionen findet für derartige Zufallsgrößen seine Anwendung in folgender Form.

Algorithmus

$$(0) \quad F_0 := 0, \quad F_i := \sum_{j=1}^i p_j \quad i=1, \dots, M$$

$$v := 0$$

(1) Würfle Z in $[0,1)$ gleichverteilt

$$(2) \quad v := v+1$$

(3) Falls $Z > F_v$ zurück zu (2)

$$\text{Sonst} \quad X := x_v$$

ENDE

Geometrische Methode

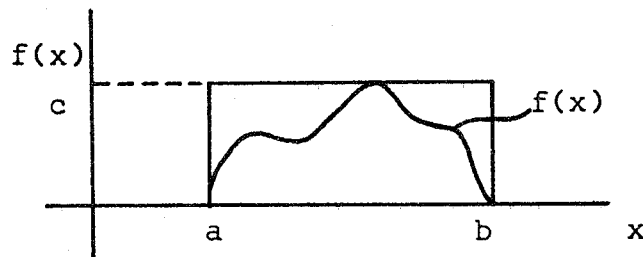
Diese Methode kann nur verwendet werden, falls die Dichtefunktion $f(x)$, also

$$f(x) = \frac{dF(x)}{dx} \quad (3-13)$$

existiert und außerdem gilt, daß

$$f(x) = 0 \text{ für } x \notin [a, b] \quad (3-14)$$

Wegen (3-14) läßt sich nun $f(x)$ in ein Rechteck einbetten, dargestellt



sodaß $\max_{x \in [a, b]} f(x) = c$.

Erzeugen wir nun Paare von Zufallszahlen r_1, r_2 , sodaß r_1 gleichverteilt in $[a, b]$ und r_2 gleichverteilt in $[0, c]$ und testen wir jeweils, ob

$$r_2 \leq f(r_1). \quad (3-15)$$

Ist (3-15) gültig, so nehmen wir r_1 an, andernfalls verwerfen wir das Paar (r_1, r_2) . Es ist klar, daß damit die Folge akzeptierter Zufallszahlen r_1 nach $f(x)$ verteilt ist.

Bestimmen wir die Wahrscheinlichkeit, daß eine gleichverteilte Zufallszahl r_1 , als nach $f(x)$ verteilt, akzeptiert wird. Dies ist, wegen der Unabhängigkeit, für ein bestimmtes x

$$P\{r_1 = x\} \cdot P\{r_2 \leq f(x)\}$$

Nun ist aber

$$P\{r_1=x\} = \frac{dx}{b-a}$$

und

$$P\{r_2 \leq f(x)\} = P\left\{\frac{r_2}{c} \leq \frac{f(x)}{c}\right\} = \frac{f(x)}{c}$$

da $\frac{r_2}{c}$ gleichverteilt in $[0,1]$ ist.

Daraus folgt für alle $x \in [a,b]$

$$P\{\text{erfolgreiches Paar } r_1, r_2\} = \int_a^b \frac{dx}{b-a} \frac{f(x)}{c} = \frac{1}{c(b-a)} \quad (3-16)$$

Ist also $c(b-a)$ sehr groß, so wird die geometrische Methode sehr ineffizient.

Grundsätzlich sind mit den angegebenen Verfahren alle eindimensionalen Verteilungen erzeugbar. Im Falle mehrdimensionaler Verteilungen, also jener, bei denen verschiedene Zufallsvariable eine gemeinsame Verteilungsfunktion $F(X_1, X_2, \dots, X_n)$ besitzen, ist die Sache nicht mehr so einfach. Nur die geometrische Methode läßt sich verallgemeinern, die beiden anderen nicht, da keine Inverse bei einer mehrdimensionalen Verteilungsfunktion existiert.

Zum Abschluß seien noch zwei spezielle Methoden zum Erzeugen Normal- bzw. Poissonverteilter Zufallszahlen erwähnt.

Normalverteilte Zufallszahlen kann man auf Grund des zentralen Grenzwertsatzes durch Summe mehrerer gleichverteilter Zufallszahlen erzeugen. Der zentrale Grenzwertsatz lautet nämlich:

Seien X_i , $i=1,2,\dots$, unabhängige Zufallsvariable, die alle die gleiche Verteilung haben und deren Momente $E[X_i] = \mu$ und $\text{Var}[X_i] = \sigma^2$ endlich sind.

Sei $S_n = \sum_{i=1}^n X_i$, dann gilt:

$$P\left(\frac{S_n - n\mu}{\sqrt{n} \cdot \sigma} \leq x\right) \xrightarrow{n \rightarrow \infty} \Phi(x)$$

wo $\Phi(x)$ die Verteilungsfunktion der Normalverteilung ist.

D.h. der Ausdruck $\frac{S_n - n\mu}{\sqrt{n} \cdot \sigma}$ strebt gegen eine Normalverteilung.

Durch Wahl von n läßt sich die Güte der Approximation festlegen.

Poissonverteilte Zufallszahlen lassen sich darstellen wegen der Beziehung zwischen Poisson- und Exponentialverteilung. Es gilt, daß der zeitliche Abstand zwischen zwei Poissonereignissen exponentialverteilt ist.

Mehrdimensionaler Fall

Sei \vec{X} ein n -dimensionaler Zufallsvektor mit Verteilungsfunktion

$$F_{\vec{X}}(\vec{x}) = F_{\vec{X}}(x_1, \dots, x_n) \quad (3-17)$$

Sind die Komponenten von \vec{X} gegenseitig unabhängig, so läßt sich (3-17) schreiben als

$$F_{\vec{X}}(\vec{x}) = F_{X_1}(x_1) \dots F_{X_n}(x_n)$$

und man kann die Realisationen von \vec{X} komponentenweise unabhängig voneinander erzeugen.

Sind die Komponenten gegenseitig abhängig, so muß man anders vorgehen. Es sei der Fall betrachtet, wo der Zufallsvektor eine Dichtefunktion $f_{\vec{X}}(x_1, \dots, x_n)$ besitzt. Diese läßt sich schreiben

$$f_{\vec{X}}(x_1, \dots, x_n) = f_{X_1}(x_1) \cdot f_{X_2}(x_2 | x_1) \cdot f_{X_3}(x_3 | x_1, x_2) \dots f_{X_n}(x_n | x_1, \dots, x_{n-1})$$

worin $f_{X_i}(x_i | x_1, \dots, x_{i-1})$ die bedingte Dichtefunktion der i -ten

Komponente X_i von \vec{X} ist, falls die $i-1$ Komponenten X_1, \dots, X_{i-1} die Werte x_1, \dots, x_{i-1} angenommen haben. Es gilt

$$f_{X_i}(x_i | x_1, \dots, x_{i-1}) = \frac{\int_{x_{i+1}=-\infty}^{+\infty} \dots \int_{x_n=-\infty}^{+\infty} f_{\vec{X}}(x_1, \dots, x_n) dx_{i+1} \dots dx_n}{\int_{x_i=-\infty}^{+\infty} \dots \int_{x_n=-\infty}^{+\infty} f_{\vec{X}}(x_1, \dots, x_n) dx_i \dots dx_n} \quad (3-18)$$

$i=2, \dots, n$

Mit den Randverteilungsdichten

$$f_i(x_1, \dots, x_i) = \int_{x_{i+1}=-\infty}^{+\infty} \dots \int_{x_n=-\infty}^{+\infty} f_{\vec{X}}(x_1, \dots, x_n) dx_{i+1} \dots dx_n$$

$i=1, \dots, n-1$

wird (3-18):

$$f_{X_i}(x_i | x_1, \dots, x_{i-1}) = \frac{f_i(x_1, \dots, x_i)}{f_{i-1}(x_1, \dots, x_{i-1})}$$

$i=2, \dots, n$

Das allgemeine Verfahren zur Erzeugung einer Realisation des Zufallsvektors \vec{X} besteht nun in folgendem

Algorithmus

(0) $v := 1$

(1) Erzeuge X_1 gemäß der Randverteilung $f_{X_1}(x_1)$

(2) Falls $v=n$ ENDE,

(3) Sonst $v := v+1$

Erzeuge X_v gemäß der bedingten Verteilung $f_{X_v}(x_v | X_1, \dots, X_{v-1})$ und zurück zu (2).

Realisierung von mehrdimensionalen, normal-verteilten Zufallsvektoren

Für diese Zufallsgrößen eignet sich ein speziell darauf zugeschnittenes Verfahren besser als das allgemeine. Sei \vec{X} der normalverteilte Zufallsvektor mit Dichtefunktion

$$f_{\vec{X}}(\vec{x}) = \frac{1}{((2\pi)^n \cdot \det C)^{\frac{1}{2}}} e^{-\frac{1}{2} [(\vec{x}-\vec{\mu})^* C^{-1}(\vec{x}-\vec{\mu})]}$$

Es ist also die positiv-definite Matrix C seine gegebene Kovarianzmatrix mit Komponenten

$$c_{ij} = \text{Cov}(X_i, X_j) \quad i, j=1, \dots, n \Rightarrow c_{ij} = c_{ji}$$

und sein Erwartungsvektor

$$\vec{\mu} = E\vec{X}$$

Das Verfahren besteht nun darin, daß ein Zufallsvektor $\vec{U}=(U_1, \dots, U_n)^*$ erzeugt wird, mit U_i normalverteilt und

$$EU_i = 0, \quad \text{Var } U_i = 1 \quad i=1, \dots, n$$

$$\text{Cov}(U_i, U_j) = \begin{cases} 1 & i=j \\ 0 & i \neq j \end{cases} \quad i, j=1, \dots, n \quad (3-19)$$

Dieses wird nun einer Linear-Transformation unterworfen

$$\vec{X} = A \cdot \vec{U} + \vec{\mu} \quad (3-20)$$

Kraft des Additionssatzes der Normal-Verteilung, besitzt sie \vec{X} ebenfalls, ferner hat \vec{X} bereits den richtigen Erwartungswert. Es muß noch A derart bestimmt werden, daß \vec{X} die gewünschte Kovarianzmatrix aufweist.

Es gilt

$$C = E[(\vec{X}-\vec{\mu})(\vec{X}-\vec{\mu})^*]$$

woraus mit (3-19) und (3-20) folgt

$$\underline{C} = E(\underline{A} \cdot \underline{U} \cdot \underline{U}^* \cdot \underline{A}^*) = \underline{A} \cdot E(\underline{U} \cdot \underline{U}^*) \cdot \underline{A}^* = \underline{A} \cdot \underline{A}^* \quad (3-21)$$

Gleichung (3-21) läßt sich im allgemeinen nicht eindeutig nach A auflösen. Folgender Ansatz führt jedoch zum Ziel: man wählt für A eine Dreiecksmatrix, dies ist möglich, da C symmetrisch ist

$$A = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ a_{21} & a_{22} & 0 & \dots & 0 \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

nun kann man (3-21) rekursiv nach den einzelnen Elementen von A lösen:

Gleichung (3-21) schreibt sich ausführlich

$$\begin{bmatrix} a_{11} & 0 & 0 & \dots & 0 \\ a_{21} & a_{22} & 0 & \dots & 0 \\ a_{31} & a_{32} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} a_{11} & a_{21} & a_{31} & \dots & a_{n1} \\ 0 & a_{22} & a_{32} & \dots & a_{n2} \\ 0 & 0 & \dots & a_{n3} \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a_{nn} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{21} & c_{31} & \dots & c_{n1} \\ c_{21} & c_{22} & \dots & \dots & \dots \\ c_{31} & \dots & \dots & \dots & \dots \\ \vdots & \dots & \dots & \dots & \dots \\ c_{n1} & \dots & \dots & \dots & c_{nn} \end{bmatrix}$$

man liest daraus ab:

$$a_{11}^2 = c_{11}$$

$$a_{22}^2 = c_{22} - a_{21}^2$$

$$a_{21} = \frac{c_{21}}{a_{11}}$$

$$a_{32} = \frac{1}{a_{22}} (c_{32} - a_{31} \cdot a_{21})$$

$$\vdots$$

$$a_{n1} = \frac{c_{n1}}{a_{11}}$$

usw.

Allgemein gilt:

Berechne zunächst

$$a_{11} := \sqrt{c_{11}}$$

$$a_{i1} := \frac{c_{i1}}{a_{11}} \quad i=2, \dots, n$$

anschließend nacheinander für $i=2, \dots, n$

$$a_{ii} := c_{ii} - \sum_{k=1}^{i-1} a_{ik}^2$$

$$a_{ki} := \frac{1}{a_{ii}} \left(c_{ki} - \sum_{\ell=1}^{i-1} a_{k\ell} a_{i\ell} \right), \quad k=i+1, \dots, n$$

3.5. Das Computerprogramm

Mit den Kenntnissen über Zufallszahlen sollte es nun möglich sein, eine Subroutine zu schreiben, welche, ausgehend von einem Kartenpaket mit 260 Karten, zufällig Karten austeilt und, falls zu Spielbeginn weniger als 50 Karten vorhanden sind, alle 260 Karten wieder neu mischt. Die entsprechende Subroutine lautet FUNCTION IZUF(KX,KY) und ist im Anhang zu Kapitel 4 gelistet.

Der nächste Schritt wird sein, eine Subroutine, welche das Spielverhalten des Croupiers bzw. des Spielers simuliert, zu schreiben. Als Input werden die bereits ausgeteilten Karten eingegeben, als Output die Antwort, ob der Croupier bzw. Spieler noch eine Karte möchte. Wichtig ist hier festzuhalten, daß beim Erhalt einer As nicht sofort deren Wert (1 oder 11) festgelegt werden muß, sondern dieser Wert dadurch bestimmt wird, daß der Spieler keine weitere Karte mehr wünscht. Dann wird für den Spieler die größte Kartensumme angenommen, die, sofern möglich unter 21 liegt. Hat also der Spieler durch eine As bei Spiel-

abbruch die Kartensumme 7 oder 17, so wird 17 angenommen, hat er 12 oder 22, so wird 12 angenommen. Für den Croupier gilt, daß er stets, auch mit As, aufhört weitere Karten zu nehmen, falls er eine Kartensumme größer als 16, aber kleiner als 22 erreicht hat. Hat also der Croupier durch eine As die Kartensumme 12 oder 22, spielt er weiter, nimmt daher die Kartensumme 12 an, bei 7 oder 17 hört er auf.

Sei $E(J)$ die Kartensumme des Spielers J (für Croupier gelte $J = 1$), $G(J) = 1$ signalisiere eine As bei Spieler J , mit $F(J) = 0$ soll der Wunsch des Spielers nach einer weiteren Karte angegeben werden und $F(J) = 2$ zeige an, daß Spieler J einen Black Jack hat.

Weiters werde mit $H(J)$ diejenige Kartensumme angegeben, bis zu der Spieler J noch Karten nimmt (es gilt daher $H(1)=16$), sowie $D(J)$ gebe diejenige Kartensumme an, ab der man eine As als 11 annimmt (also z.B. $D(1) = 17$). Dann lautet die entsprechende Subroutine

SUBROUTINE SP(J,E,F,G,H,D) - siehe Anhang Kapitel 4.

Nunmehr muß eine Subroutine erstellt werden, welche den Spielablauf beschreibt sowie die Gewinne für jeden Spieler berechnet. Sei $A(J)$ der Gewinn bzw. Verlust von Spieler J , falls er mit Einsatz von Betrag 1 spielt. Sei N die Anzahl Spieler ($N \leq 7$). Dann lautet das Programm

SUBROUTINE SPIEL (A,N,KX,H,D)

Sind wir soweit, können wir ein Spiel vollständig durchspielen bzw. beliebig oft wiederholen. Da es sich ja um ein stochastisches Modell handelt, sind die Resultate eines Spiels nicht sehr aussagefähig. Statt die Gewinne eines Spiels zu betrachten, interessieren uns vielmehr die erwarteten Gewinne bzw. Verluste nach einer Reihe von Spielen. Wir müssen also die Erwartungswerte schätzen. Die Formel hierfür ist bekannt. Sei G_j der Gewinn (Verlust) eines beliebigen Spielers im Spiel j und werden

M Spiele gespielt, so gilt für den erwarteten Gewinn E_G

$$E_G = \frac{\sum_{j=1}^M G_j}{M} \quad (3-22)$$

Damit lautet nun das Programm, welches auch das Abbruchkriterium überprüft (\rightarrow siehe 3.6.)

SUBROUTINE BJ(H,D,N,L,E,A).

Alle erwähnten Programmanteile sind im Anhang zu Kapitel 4 aufgelistet.

3.6. Problem II: Wie oft simulieren?

Bei jedem stochastischen Simulationsmodell erhebt sich die Frage, wie oft derselbe Prozeß simuliert werden soll. Die grundsätzliche Antwort darauf ist: So oft, bis die statistischen Kenngrößen (wie Erwartungswert, Varianz, Verteilungsfunktion) von stochastischen Variablen mit ausreichender statistischer Sicherheit geschätzt werden können. Welche Fehlerwahrscheinlichkeit als ausreichend bzw. zu hoch angesehen wird, muß natürlich vom Modellbauer vorgegeben werden. Das Problem ist eine Standardfrage der Statistik, nämlich, ob eine gewisse Stichprobe bestimmte Schlüsse zuläßt, oder ob noch mehr Stichprobenresultate erforderlich sind. Wir wollen und hier nur mit der Frage beschäftigen, wie die Sicherheit geschätzter Erwartungswerte bestimmt werden kann.

Es werde also der Erwartungswert einer Zufallszahl gemäß (3-22), nämlich

$$\bar{X} = \frac{\sum_{j=1}^n X_j}{n}$$

geschätzt. Nun ist aber \bar{X} eine Funktion von Realisierungen einer Zufallszahl und demnach selber zufällig. Sind die

einzelnen Stichproben X_j unabhängig voneinander und mit dem Erwartungswert μ und der Varianz σ^2 verteilt, so gilt

a) $E(\bar{X}) = \mu$

b) $\text{Var}(\bar{X}) = \frac{\sigma^2}{n}$

c) Für große n ist $\frac{\bar{X} - \mu}{\frac{\sigma}{\sqrt{n}}}$ approximativ normal verteilt.

Beweis von b)

$$\text{Var}(\bar{X}) = \text{Var}\left(\frac{1}{n} \sum_{j=1}^n X_j\right) = \frac{1}{n^2} \sum_{j=1}^n \text{Var}(X_j) = \frac{\sigma^2}{n}$$

Wir können nun auch σ^2 schätzen, nämlich gemäß der Formel (hiebei wird $n - n-1$ angenommen, also $n \gg 1$)

$$S^2 = \frac{1}{n} \sum_{j=1}^n (X_j - \bar{X})^2 = \frac{1}{n} \sum_{j=1}^n X_j^2 - \bar{X}^2 \quad (3-23)$$

Approximiert man nun σ^2 durch S^2 , so läßt sich die Varianz von \bar{X} schätzen durch

$$\text{Var}(\bar{X}) \sim \frac{S^2}{n} \quad (3-24)$$

$\sqrt{\frac{S^2}{n}}$ liefert daher ein Maß für die Abweichung des geschätzten Erwartungswertes vom wirklichen Erwartungswert und man gibt als Abbruchkriterium eine Größe $\epsilon(\bar{X}) > 0$ vor, sodaß das Computerprogramm abgebrochen wird, falls gilt

$$\sqrt{\frac{S^2}{n}} \leq \epsilon(\bar{X}) \quad (3-25)$$

Hiebei ist $\epsilon(\bar{X})$ ein Maß für den absoluten Fehler des geschätzten Erwartungswertes. $\epsilon(\bar{X})$ soll besagen, daß das Abbruchkriterium vom Wert \bar{X} abhängen kann, z.B. $\epsilon(\bar{X}) = K \cdot |\bar{X}|$, oder aber einen festen Wert $\epsilon(\bar{X}) = K$ haben kann. In der Subroutine BJ(H,D,N,L,E,A) wurde $\epsilon(X) = K \cdot |\bar{X}|$ gewählt.

3.7. Literatur und Übungsaufgaben

Literatur:

Knuth, D.E., The Art of Computer Programming, Volume 2,
Seminumerical Algorithms, Addison-Weseley, 1969

Thorp, E., Beat the Dealer, Random House, 1966

Liebling, Th.M., Simulation, Vorlesungsunterlagen,
Institut für Operations Research, ETH-Zürich, 1973

Epstein, R., Theory of Gambling and Statistical Logic
Academic Press, 1967

Aufgaben:

Aufgabe 1: Erzeugung von Zufallsvektoren

Sie verallgemeinern zunächst das
Verfahren der geometrischen Wahrscheinlichkeiten, sodaß damit
n-dimensionale Zufallsvektoren mit gegebener Wahrscheinlich-
keitsdichte $f(x_1, \dots, x_n)$ erzeugt werden können.

Dabei geben Sie die Voraussetzungen an, unter denen sich das
Verfahren anwenden läßt. Anschließend bestimmen Sie die
Verteilung der Anzahl v Iterationszyklen, die benötigt werden,
um einen Zufallsvektor zu erhalten, insbesondere den Erwartungs-
wert von v .

Schließlich wenden Sie (von Hand) das Verfahren auf die
spezielle Dichte

$$f(x_1, x_2) = \begin{cases} 2(x_1 + x_2) & : 0 < x_1 < x_2 \leq 1 \\ 0 & \text{sonst} \end{cases}$$

an.

Verwenden Sie dabei die Zufallszahlen

$$Z_1 = 3756, \quad Z_2 = 8765, \quad Z_3 = 3671$$

wobei allgemein

$$0 \leq Z_i \leq 9999$$

gelten soll.

Aufgabe 2: Erzeugung von Pseudozufallszahlen

(a) Lineare Kongruenzen Methode

Wählt man die ganzen Zahlen

x_0 (Anfangswert ; $x_0 \geq 0$)

a (Multiplikator; $a \geq 0$)

c (Inkrement ; $c \geq 0$)

m (Modulus ; $m > x_0, m > a, m > c$)

geschickt, so entsteht mit Hilfe der Relation

$$x_{n+1} = (a \cdot x_n + c) \bmod m, n \geq 0$$

eine Zahlenfolge, die gewisse Eigenschaften aufweisen soll, welche man von einer Folge von zufälligen Ziehungen von ganzen Zahlen zwischen 0 und $m-1$ erwarten würde.

Bei genauerer Untersuchung zeigt es sich jedoch, daß, abgesehen von ihrem periodischen Charakter, die "Zufälligkeit" der so erzeugten Folgen auch lokal fragwürdig sein kann.

Das weiter unten angegebene Verfahren M von Mac Laren und Marsaglia behebt mit relativ geringem Aufwand diese Mängel weitgehend.

(b) Methode M

Es wird von zwei verschiedenen, nach Verfahren (a) erzeugten Folgen $\{x_n\}$ und $\{Y_n\}$ von m -stelligen Zahlen ausgegangen, ferner wird ein 64-stelliges Array $V(0), \dots, V(63)$ benötigt.

Die gesuchte Folge lautet $\{Z_\lambda\} \lambda \geq 0$.

Algorithmus

(0) $V(k) := x_k$ $k=0, \dots, 63$

$v := k+1$

$\lambda := 0$

(1) $x := x_v$

$Y := Y_\lambda$

$J := \text{entier} \left(\frac{k}{m} \cdot Y \right)$

$Z_\lambda := V(J)$

$V(J) := x$

(2) $v := v+1$

$\lambda := \lambda+1$

zurück zu Schritt (1), um neue Zufallszahlen Z_λ zu erzeugen.

Sie sollen ein Computer-Programm zum funktionieren bringen, das nach Methode M Zufallszahlen $Z_v, v \geq 0$, erzeugt.

Dabei verwenden Sie für $\{x_n\}$, $\{Y_n\}, n \geq 0$:

$$x_0 = 5772156649$$

$$x_{n+1} = (3141592653 \cdot x_n + 2718281829) \bmod 2^{35}, n \geq 0$$

$$Y_0 = 17801072418$$

$$Y_{n+1} = (2718281829 \cdot Y_n + 3141592653) \bmod 2^{35}, n \geq 0$$

Aufgabe 3: Begutachtung von Pseudozufallszahlen

Die Beurteilung der "Zufälligkeit" einer Folge $\{Z_n\}$ von Pseudozufallszahlen geschieht mit Hilfe einer Test-Batterie. Darunter befindet sich der sogenannte Maximum-Test, welcher auf folgendem Prinzip basiert:

Ist $\{Z_n\}$ eine Folge gegenseitig unabhängiger im Intervall $[0,1]$ gleichmäßig verteilter Zufallsvariablen, so haben die Glieder der Folge $\{M_k\}$ $k=0,1,\dots$

$$\begin{aligned} M_0 &= \max (Z_1, \dots, Z_n) \\ M_1 &= \max (Z_{n+1}, \dots, Z_{2n}) \\ &\vdots \\ M_k &= \max (Z_{k \cdot n + i} \mid i=1, \dots, n) \\ &\vdots \end{aligned}$$

die Verteilungsfunktion

$$P\{M_k \leq z\} = z^n : z \in [0,1] (*)$$

Der Maximum-Test besteht nun darin, daß man die empirische Verteilungsfunktion der Folge

$$M_k \quad k=1, \dots, K$$

mit der Verteilungsfunktion (*) vergleicht, etwa mit dem Smirnov-Kolmogorov (SK) Test.

Sie sollen für $n=3$ und $k=30$ bei einer Fehlerwkt. 1. Art von 0.95 den SK-Test auf die vom Algorithmus der Aufgabe 2 erzeugten Zahlen anwenden.

4.

OPTIMIERUNG

4.1. Einleitung

Wir sind nun soweit, die Frage anzugehen, welches die optimale Strategie für die Spieler ist. Zwei Parameter können wir frei wählen: Einerseits die Kartensumme, bis zu der wir weitere Karten nehmen, andererseits die Kartensumme, bei der wir den Wert einer "As" als 1 bzw. 11 annehmen. In diesem Kapitel wollen wir nun allgemein erörtern, wie man optimale Werte der kontrollierbaren Variablen bei Simulationsmodellen findet. Die dabei zur Anwendung gelangenden Methoden sind Suchalgorithmen, welche aus dem Gebiet der nichtlinearen Optimierungstheorie bekannt sind.

Ziel eines Versuchsplanes eines Simulationsmodells ist ja die beste (oder annähernd die beste) Lösung des Problems mit minimalen erwarteten Computerkosten zu finden. Derzeit existiert kein Verfahren um solch einen optimalen Versuchsplan zu erstellen. Insofern sind die folgenden Erläuterungen nur als nützliche Hilfen beim Planen einer Versuchsreihe zu verstehen.

Zunächst sind in einem Simulationsmodell diejenigen Variablen zu identifizieren, deren Werte vom Versuchsplaner gesteuert werden können, um ein Optimum zu finden (sogenannte kontrollierbare Variablen). Der Bereich in dem diese Variablen einzeln oder in Kombination mit anderen variiert werden können, ist sodann zu bestimmen. Je enger dieser Bereich gefaßt werden kann, um so einfacher gestaltet sich die Suche eines Optimums. Wichtig ist die Unterscheidung in diskrete kontrollierbare Variable und kontinuierliche. Bei diskreten Variablen kann man im Prinzip alle möglichen Werte absuchen und so das Optimum finden. Allerdings ist diese Vorgangsweise sehr aufwendig. Bei kontinuierlichen Variablen ist aber auch diese Suchweise unmöglich, da es ja unendlich viele Werte gibt. Mathematisch geschrieben sieht unser Problem so aus: Das Simulationsmodell besteht aus einer Menge von kontrollierbaren Variablen

X_1, X_2, \dots, X_n . Diese Variablen können nur Werte aus einem vorgegebenen Bereich B annehmen, also

$$\{X_1, \dots, X_n\} \in B \subset \mathbb{R}^n \quad (4-1)$$

Weiters liefert das Simulationsmodell bei einem gegebenen Input $\{X_1, \dots, X_n\}$ einen Output, welcher das Maß für die Güte der gefundenen Lösung darstellt, deren Wert optimiert, also maximiert oder minimiert werden soll.

Also

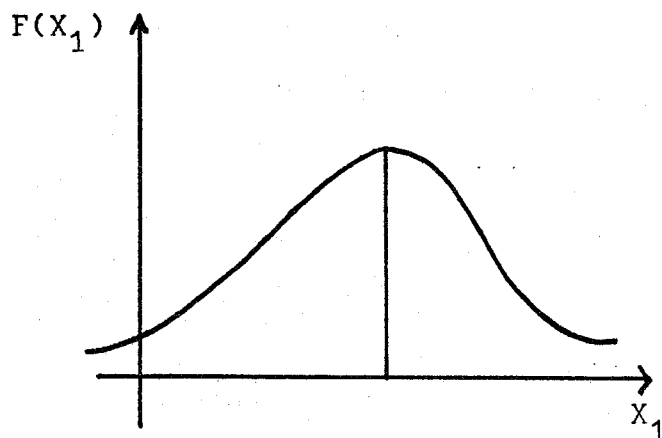
$$\begin{aligned} F: \mathbb{R}^n &\rightarrow \mathbb{R} \\ F(X_1, \dots, X_n) &\rightarrow \max \end{aligned} \quad (4-2)$$

Ein Problem, wie in (4-1) und (4-2) definiert heißt nicht-lineares Optimierungsproblem. Gilt für X_1, \dots, X_n noch zusätzlich, daß diese nur diskrete Werte, z.B. ganzzahlige, annehmen dürfen, so heißt es nichtlineares ganzzahliges Optimierungsproblem. Nicht immer läßt sich ein Maß für die Güte der Lösung wie in (4-2) angeben. Oftmals sollen bei einem Problem mehrere, mitunter einander widersprechende Ziele gleichzeitig erreicht werden. Das würde bedeuten, daß man als Output eine Reihe von Kenngrößen erhält, also

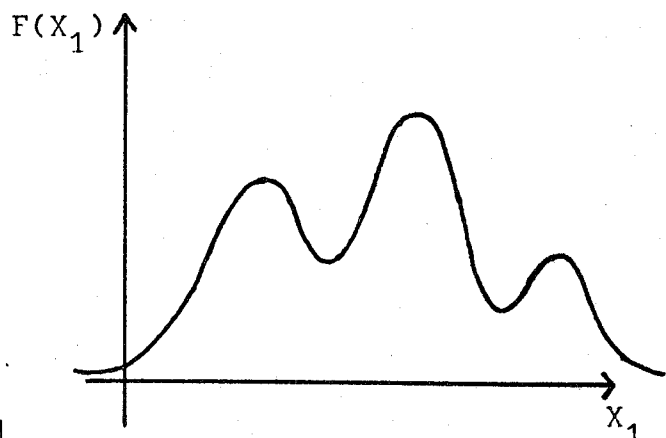
$$\begin{aligned} &F_1(X_1, \dots, X_n) \\ &\dots\dots\dots \\ &F_m(X_1, \dots, X_n) \end{aligned} \quad (4-3)$$

Diese lassen sich aber praktisch nie gleichzeitig maximieren. Es ist dann Aufgabe des Entscheidenden, durch Bewertung der einzelnen Werte F_1, \dots, F_m festzulegen, ob eine Lösung besser als eine andere ist, aber darauf wollen wir hier nicht näher eingehen, sondern uns mit Modellen gemäß (4-2) zufrieden geben.

Wichtig ist noch die Frage, ob ein Simulationsmodell nur eine oder mehrere optimale Lösungen aufweist. Was man darunter versteht, wollen wir im Falle einer einzigen kontrollierbaren Variablen graphisch klarmachen.



Eine optimale Lösung



Mehrere optimale Lösungen

Alle Suchalgorithmen haben die Eigenschaft nur eine optimale Lösung zu finden, aber, falls es mehrere gibt, nicht notwendigerweise die beste unter den optimalen Lösungen. Ein gezieltes Vorgehen bei einem Problem mit mehreren Optimallösungen existiert nicht. Man kann nur durch Starten des Suchalgorithmus von verschiedenen Punkten aus erreichen, verschiedene dieser Optimallösungen zu finden. Eine recht gute Analogie zu Suchalgorithmen wäre der Versuch eines Bergsteigers in mitten einer Gebirgslandschaft bei starkem Nebel den höchsten Berg zu finden. Er wird wohl einen Gipfel finden, aber er kann nicht sicher sein, daß es der höchste ist. Anders ausgedrückt: Suchalgorithmen können stets nur lokale, aber nicht globale Optima aufspüren. Leider existiert auch kein Verfahren mit dem man bei einem Simulationsmodell feststellen könnte, ob es ein oder mehrere Optima aufweist. Hier ist man meist ausschließlich auf die intuitive Einsicht in das Modell angewiesen.

4.2. Spezielle Suchalgorithmen

Es sollen nun einige gängige Suchalgorithmen beschrieben werden, wobei keine detaillierten mathematischen Überlegungen gebracht werden können, da dies eine eigene Vorlesung über nichtlineare Optimierung erfordern würde. Nochmals sei darauf aufmerksam gemacht, daß alle Suchalgorithmen die beste Lösung nur im Falle eines einzigen, globalen Optimums approximieren. Desgleichen sei festgehalten, daß, wiewohl einige Algorithmen auch im Falle diskreter (ganzzahliger) Variablen angewandt werden können, selbst bei Existenz nur eines globalen Optimums die Konvergenz zu diesem Optimum nicht gesichert ist. In der Tat existiert für das allgemeine nichtlineare, ganzzahlige Optimierungsproblem kein Algorithmus mit dem das Auffinden der Optimallösung gesichert wäre. Grundsätzlich läßt sich aber das Optimum durch Simulationsläufe mit allen möglichen Variablenwerten stets finden, wenngleich nur mit astronomischen Computerzeiten.

Eindimensionale Suche

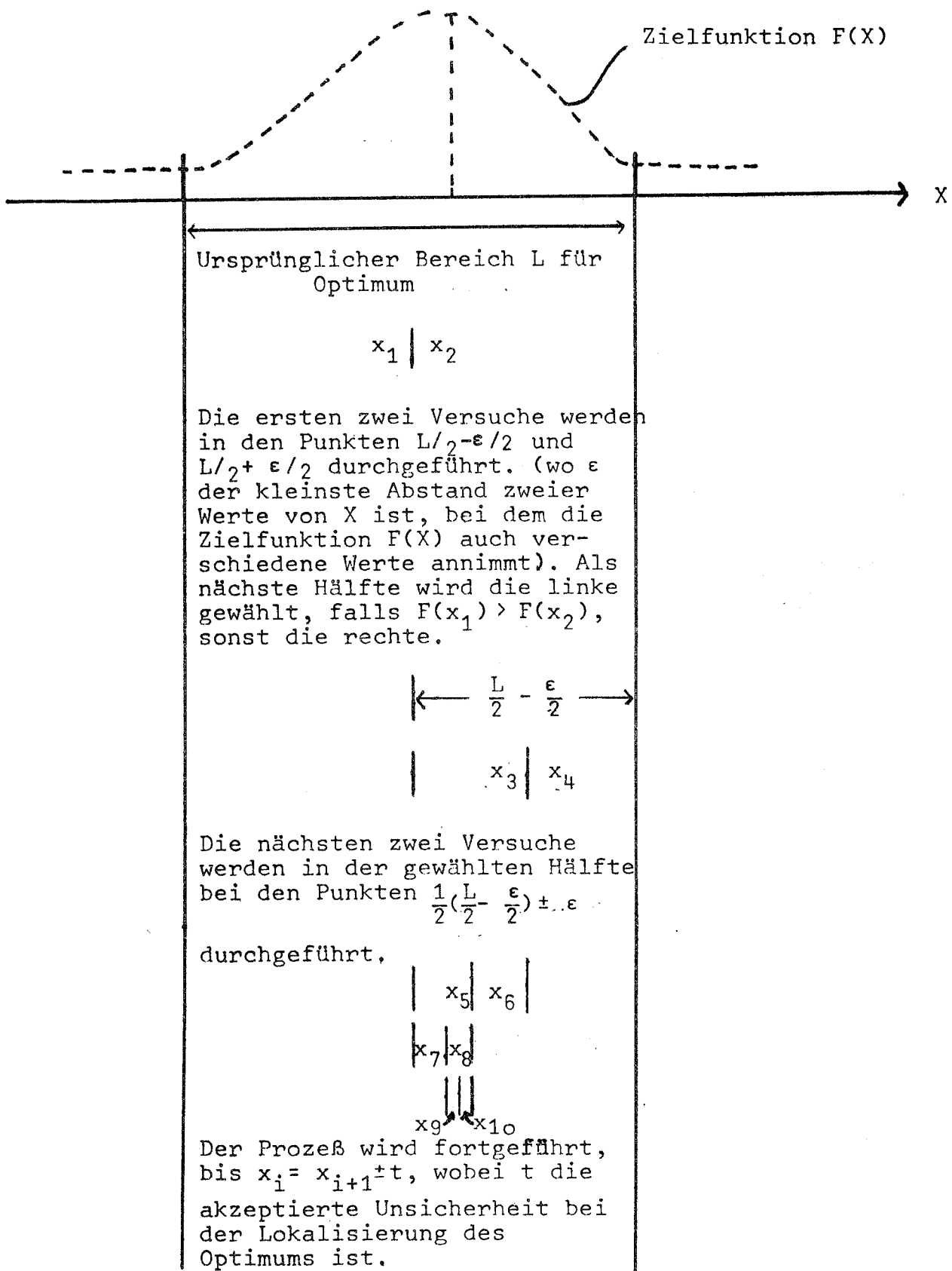
Das allgemeine Verfahren zur Suche des Optimums bei mehr als einer kontrollierbaren Variablen (mehrdimensionales Problem) besteht darin, die Suche auf eine Folge eindimensionaler Suchvorgänge zu reduzieren. Im mehrdimensionalen Problem wird stets eine Suchrichtung festgelegt und dann nur in dieser Richtung das Optimum gesucht, was dann ein eindimensionaler Suchvorgang ist.

Zwei Faktoren müssen bei der eindimensionalen Suche stets bestimmt werden:

- (1) Die Suchrichtung (vom jetzigen Punkt)
- (2) Die Schrittweite (das ist der Abstand zum nächsten Versuchspunkt).

Wählt man eine kleine Schrittweite und berechnet die Zielfunktion in Punkten welche im Abstand der gewählten Schrittweite voneinander liegen, wobei man sich in der Richtung des ansteigenden Wertes der Zielfunktion bewegt, so ist dies zwar ein sicheres Verfahren aber sehr zeitaufwendig. Viele Schritte sind nötig und in jedem Schritt muß das Simulationsmodell durchgespielt werden, um die Zielfunktion zu bestimmen.

Falls daher ein Bereich mit oberer und unterer Schranke für die Lage des Optimums angegeben werden kann, ist eine binäre Suche entschieden effektiver. Bei der binären Suche wird der Bereich halbiert und dann bestimmt, in welchen der beiden Hälften das Optimum liegt usw. Dieser Algorithmus konvergiert sehr rasch zum Optimum. Die Schrittweite des Suchprozesses wird also in jeder Iteration halbiert. Figur 4-1 zeigt den Ablauf eines binären Suchprozesses.

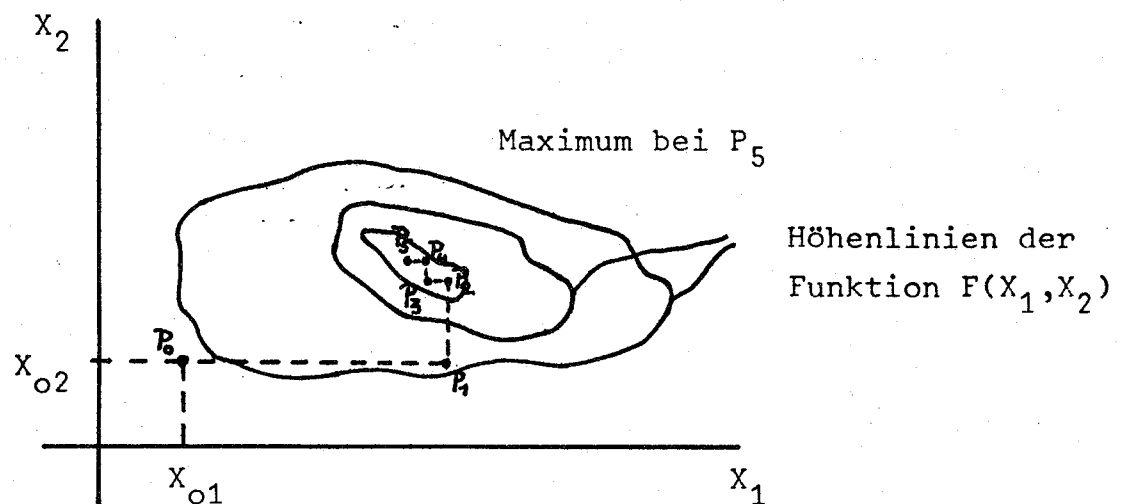


Figur 4-1

Binärer Suchprozeß

Einfaktoren-Suche

Bei dieser Methode handelt es sich um die einfachste Erweiterung des eindimensionalen Suchprozesses auf mehrdimensionale Probleme. Und zwar wird jeweils in einer Variablen, bei Konstanthaltung aller übrigen, optimiert, solange bis man den Wert der Zielfunktion durch Veränderung einer einzigen Variablen nicht mehr verbessern kann. Um dies genauer zu illustrieren, betrachten wir ein Problem in zwei Variablen X_1 , X_2 mit der Zielfunktion $F(X_1, X_2)$. Wir wählen einen Startwert für beide Variablen, nämlich X_{o1} und X_{o2} . Mit diesen Werten wird das Simulationsmodell durchgerechnet und man erhält den Wert $P_o = F(X_{o1}, X_{o2})$. Nun wird eine eindimensionale Suche in X_1 durchgeführt, bis man einen in X_1 optimalen Wert $P_1 = F(X_1, X_{o2})$ erhält. Nun wird X_1 festgehalten und eine Suche in X_2 durchgeführt usw. Figur 4-2 gibt eine geometrische Deutung dieses Verfahrens.

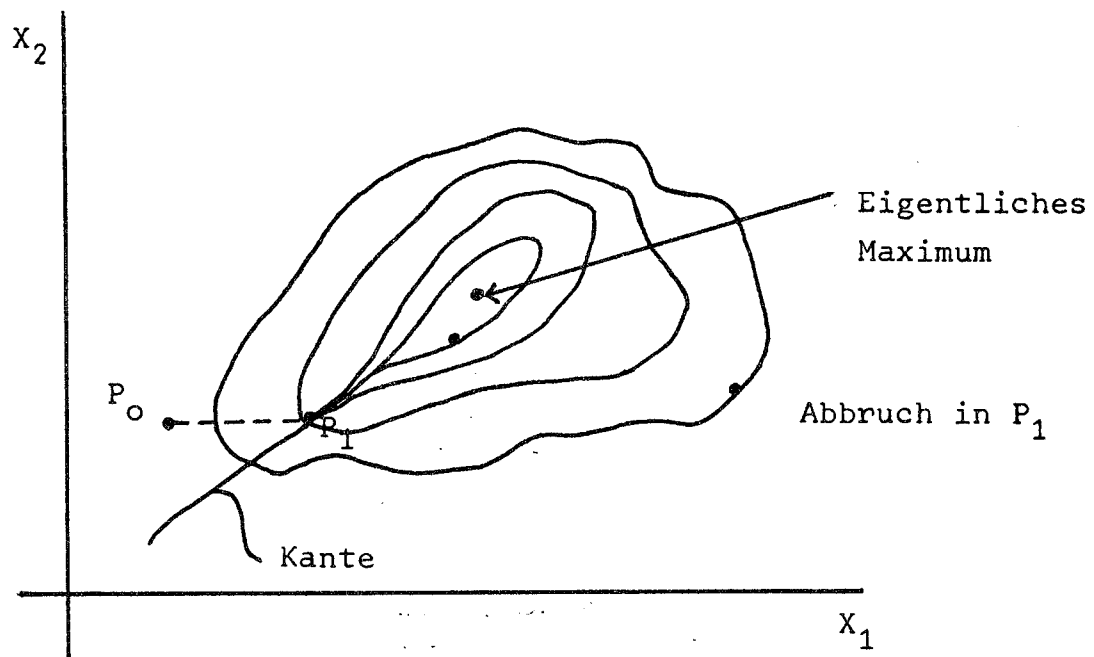


Figur 4-2

Zweidimensionales Beispiel einer Einfaktoren-Suche

Die Erweiterung der Einfaktoren-Suche auf n-dimensionale Probleme ergibt sich leicht aus dem zweidimensionalen Fall.

Der dargestellte Algorithmus weist aber zwei Schwächen auf. Zum einen müssen i.a. mehr Punkte berechnet werden, als bei Algorithmen bei denen mehrere Variablen gleichzeitig verändert werden können. Zum anderen kann die Methode bei Zielfunktionen, welche Kanten aufweist, versagen, wie in Figur 4-3 dargestellt.



Figur 4-3
Einfaktoren-Suche auf einer Fläche mit Kante

Bei der nun beschriebenen Methode treten diese Schwierigkeiten nicht auf.

Gradienten Methode

Bei diesem Algorithmus können sämtliche Variablen gleichzeitig verändert werden. Die Richtung in der der nächste Punkt gesucht wird, ist nun nicht mehr die X_1 -oder X_2 -Richtung wie in Figur 4-2

sondern jene Richtung, in der die Zielfunktionen ihren größten Wertzuwachs aufweist, also in Richtung des Gradienten der Zielfunktion.

Erläutern wir das Verfahren wieder am zweidimensionalen Fall. Gegeben sei wieder die Zielfunktion $F(X_1, X_2)$. Wir geben einen Startpunkt X_{01}, X_{02} vor und bestimmen nun den Gradienten $G = (g_1, g_2)$ in diesem Punkt. Die Komponenten von G sind definiert als

$$g_i = \left. \frac{\partial F(X_1, X_2)}{\partial X_i} \right|_{X_i = X_{0i}}, \quad i=1,2.$$

Diese partiellen Ableitungen werden approximiert durch

$$g_1 \sim \frac{F(X_{01} + \delta, X_{02}) - F(X_{01}, X_{02})}{\delta} \quad (4-4)$$

$$g_2 \sim \frac{F(X_{01}, X_{02} + \delta) - F(X_{01}, X_{02})}{\delta}$$

wobei $\delta > 0$ eine beliebige kleine Zahl ist.

Der Gradientenvektor $G = (g_1, g_2)$ bestimmt nun die Suchrichtung, d.h. ändert man die Variable X_1 um

$$X_{01} + Y \cdot g_1, \quad Y > 0 \quad \text{beliebig} \quad (4-5)$$

so muß sich X_2 ändern um

$$X_{02} + Y \cdot g_2.$$

Hat man bei einer eindimensionalen Suche gemäß (4-5) ein Optimum gefunden, so beginnt man wieder mit Berechnung von (4-4). Wichtig ist, daß die Suche stets in Richtung des

positiven Gradienten erfolgt. Dies wird erreicht, indem in (4-5) stets $Y > 0$ gelten muß. Das Verfahren bricht wiederum ab, falls man bei der eindimensionalen Suche keinen besseren, als den bereits gefundenen Wert, erhält. Gradientenverfahren garantieren aber nur Konvergenz zu einem Punkt mit $\text{grad } F=0$, was sowohl ein Optimalpunkt, als auch ein Sattelpunkt sein kann.

Neben den angeführten Methoden existieren noch viele andere, welche jeweils bei bestimmten Zielfunktionen besonders effizient arbeiten. Allerdings kennt man die Struktur der Zielfunktion bei Simulationsmodellen nicht, weswegen ein a priori Entscheid für den einen oder anderen Suchalgorithmus kaum unter dem Aspekt der Effizienz gefällt werden kann.

4.3. Optimierung bei Black Jack

Im Falle unseres konkreten Problems wollen wir die binäre Suche im eindimensionalen Fall verwenden und, da wir ja zwei Variablen steuern können, diese mit der Einfaktoren-Suche kombinieren. Hierbei ist es noch wichtig den Bereich der eindimensionalen Suche abzugrenzen.

Der für eine optimale Strategie relevante Bereich läßt sich dadurch einschränken, daß man mit einer Kartensumme von 11 (egal, ob man eine As besitzt oder nicht) sicher noch eine Karte nimmt und bei einer Kartensumme von 21 sicher keine Karte mehr nimmt.

Das komplette Computerprogramm zur Bestimmung einer optimalen Strategie ist im Anhang aufgelistet.

Die Güte der so erhaltenen Optimalstrategie hängt natürlich sehr von der Richtigkeit der Schätzung der Verlustwahrscheinlichkeit bei den einzelnen Strategien ab, damit also von dem Abbruchparameter ϵ . Wählt man ϵ kleiner so wird die Lösung exakter. Dabei kann sich die Optimalstrategie ändern oder auch nicht. Um darüber genauere Information zu erhalten muß man das Programm für verschiedene Werte von ϵ durchrechnen.

Wir haben nun die Verhaltensstrategie des Spielers abhängig gemacht von seiner eigenen Kartensumme. Nun ließe sich eine komplizierte Strategie auch abhängig von der ersten Karte des Croupiers berechnen, indem man im Programm dem Croupier stets die gleiche erste Karte z.B. stets eine Sieben gibt und dann die Optimalstrategie für den Spieler berechnet. Dies ist mit dem vorhandenen Programm leicht zu bewerkstelligen. Man erhält dann also zehn verschiedenen Strategien je nach erster Karte des Croupiers die natürlich zumindestens keine größere Verlustwahrscheinlichkeit für den Spieler bedeuten, als die ursprüngliche Strategie. Führt man diese Rechnung durch, so erhält man folgende Strategien:

1. Karte des Croupiers	1= =As	2	3	4	5	6	7	8	9	10
Spielabbruch d. Spielers bei Kartensumme (ohne As)	17	14	12	12	12	13	16	16	16	16
Spielabbruch d. Spielers bei Kartensumme (mit As)	17	17	15	19	19	18	18	17	17	17
Erwarteter Gewinn	-0.3121	0.0660	0.0887	0.1311	0.1539	0.1832	0.1204	0.0414	-0.0506	-0.1722

Daraus ergibt sich ein mittlerer Gewinn von -0.0205, womit man, wie zu erwarten war, besser abschneidet, als mit einer einzigen Strategie unabhängig von der ersten Karte des Croupiers. Für diesen Fall lautet das Resultat (siehe Seite 63):

Spielabbruch des Spielers bei Kartensumme (ohne As)	15
Spielabbruch des Spielers bei Kartensumme (mit As)	18
Erwarteter Gewinn	-0.03517

Weitere Verbesserungen der Strategie erzielt man durch Berücksichtigen bereits gezogener Karten, also indem man die aktuelle Wahrscheinlichkeitsverteilung für die einzelnen Karten kalkuliert. Genauer wollen wir hier nicht darauf eingehen, verweisen hiefür auf das Buch von E.Thorp.

4.4. Literatur

Zangwill, W.I., Nonlinear Programming, a unified approach,
Prentice Hall, Englewood Cliffs, N.J. (1969)

4.5. Anhang: Das Computerprogramm

```
C      PROGRAMM ZUM AUFFINDEN EINER OPTIMALSTRATEGIE BEI BLACK JACK
C      WICHTIGE GROESSEN IM PROGRAMM:
C      N ANZAHL DER SPIELER
C      E RELATIVES FEHLERMASS FUER ABBRUCHKRITERIUM
C      H (J) SPIELSTRATEGIE FUER SPIELER J
C      D (J) SPIELSTRATEGIE FUER SPIELER J MIT AS
C      S (J) SCHRANKEN FUER MOEGLICHE STRATEGIEN
C      W (J) AKTUELLE OPTIMALSTRATEGIE
C      INTEGER H(8),D(8),S(2),R(2),Q(2),X(2)
C      DIMENSION A(8),F(2),W(3)
C      EINLESEN DER PARAMETER L, R(1), R(2), E
210    READ 210,L,R(1),R(2),E
C      FORMAT(I4,2I2,F4.2)
C      WERTZUWEISUNG
C      N=2
C      H(1)=16
C      D(1)=17
C      W(3)=-100000.
C      X(1)=0
C      W(1)=R(1)
C      X(2)=0
C      W(2)=R(2)
C      IL=2
C      S(1)=11
C      S(2)=22
C      BERECHNUNG ZWEIER BENACHBARTER STRATEGIEN
15    H(2)=R(1)
C      D(2)=R(2)
C      CALL BJ(H,D,N,L,E,A)
C      F(1)=A(2)
50    IF(R(IL)+1.GE.S(2)) GOTO 105
C      Q(IL)=R(IL)+1
C      Q(3-IL)=R(3-IL)
C      H(2)=Q(1)
C      D(2)=Q(2)
C      CALL BJ(H,D,N,L,E,A)
C      F(2)=A(2)
C      FESTLEGUNG DER NAECHSTEN UNTERSUCHTEN STRATEGIEN
C      CALL SUCH(F,S,W,R,IL)
C      IF(R(IL).EQ.S(1)) GOTO 100
C      VERAENDERUNG DER SUCHRICHTUNG
C      GOTO 15
105   IF(F(1).LE.W(3)) GOTO 100
C      W(3)=F(1)
C      W(IL)=R(IL)
C      ABFRAGE, OB NOCH WEITER GESUCHT WERDEN SOLL
100   IF(IFIX(W(1)).EQ.IFIX(X(1)).AND.IFIX(W(2)).EQ.IFIX(X(2))) GOTO 200
C      IF(IL.EQ.1) GOTO 5
C      GOTO 10
C      FESTLEGEN DER PARAMETER IN DER NEUEN SUCHRICHTUNG
5     S(1)=11
C      S(2)=22
C      GOTO 30
10    S(1)=10
C      S(2)=21
```

```

30  X(1)=W(1)
    X(2)=W(2)
    R(IL)=W(IL)
    IL=3-IL
    IF(2#W(IL)-S(1)-S(2)) 35,35,40
35  R(IL)=W(IL)
    F(1)=W(3)
    GOTO 50
40  R(IL)=W(IL)-1
    GOTO 15
C   DRUCKEN DES RESULTATS
200 PRINT 205,(W(K),K=1,3)
205 FORMAT('ODIE OPTIMALSTRATEGIE LAUTET      H=',F3.0,'      D=',F3.0/'
X     MIT EINER VERLUSTWAHRSCHEINLICHKEIT VON P=',F8.5)

```

```

C   SUBROUTINE SUCH(F,S,W,R,I)
C   PROGRAMM ZUM AUFFINDEN EINER OPTIMALEN EINDIMENSIONALEN STRATEGIE
C   WICHTIGE GROESSEN IM PROGRAMM
C   I SUCHRICHTUNG (=1 ODER 2)
C   WEITERE GROESSEN SIEHE HAUPTPROGRAMM
    INTEGER S(2),R(2)
    DIMENSION F(2),W(3)
C   BESTIMMUNG DER SUCHRICHTUNG
    IF(F(1)-F(2)) 5,5,10
5   S(1)=R(I)+1
    IF(F(2).LE.W(3)) GOTO 6
    W(3)=F(2)
    W(I)=R(I)+1
6   GOTO 15
10  S(2)=R(I)
    IF(F(1).LE.W(3)) GOTO 15
    W(3)=F(1)
    W(I)=R(I)
C   FESTLEGEN DER NAECHSTEN UNTERSUCHTEN STRATEGIEN
15  R(I)=INT((S(2)-S(1))/2.)+S(1)
    RETURN

```

```

C   SUBROUTINE BJ(H,D,N,L,E,A)
C   PROGRAMM ZUM SCHAETZEN DER VERLUSTWAHRSCHEINLICHKEIT BEI FESTGELEG
C   TER STRATEGIE.
C   WICHTIGE GROESSEN (SOWEIT NICHT IM HAUPTPROGRAMM ERWAEHNT)
C   JX ANZAHL DURCHGEFUEHRTER SPIELE
C   KX SIEHE FUNCTION IZUF
C   B (J) SUMME DER SPIELAUSZAHLUNG FUER SPIELER J
C   A (J) ERWARTETE SPIELAUSZAHLUNG FUER SPIELER J
C   V (J) STANDARDABWEICHUNG VON A (J) FUER SPIELER J
    INTEGER H(8),D(8),KX(16)
    DIMENSION A(8),B(8),C(8),V(8)
    PRINT 105,N,L,E
    PRINT 120,(K,H(K),D(K),K=1,N)
    KX(14)=0
    DO 5 J=1,N
    B(J)=0
5   C(J)=0
    JX=0

```

```

25  JX=JX+1
C   DURCHFUEHREN EINES SPIELS
    CALL SPIEL(A,N,KX,H,D)
    DO 15 K=1,N
      B(K)=B(K)+A(K)
15  C(K)=C(K)+A(K)*A(K)
    IF(MOD(JX,L).NE.0) GOTO 25
    DO 20 K=1,N
C   SCHAETZEN DES ERWARTUNGSWERTES
      A(K)=B(K)/JX
20  V(K)=SQRT((C(K)/JX-A(K)*A(K))/JX)
    PRINT 125,JX,(A(K),V(K),K=1,N)
C   TEST DES ABRUCHKRITERIUMS
    DO 30 K=1,N
30  IF(V(K).GT.E*ABS(A(K))) GOTO 25
100 FORMAT(I1,I4,F6.4)
105 FORMAT('OANZAHL SPIELER =',I1,'      JEDES ',I4,'-TE SPIEL WIRD AUSG
XEDRUCKT '/' ZUGELASSENER RELATIVER FEHLER =',F6.4)
110 FORMAT(16I2)
120 FORMAT(36H DIE VERHALTENSPARAMETER FUR SPIELER,I2,7H LAUTEN,4H  H=
X,I2,4H  D=,I2)
125 FORMAT(19H DIE WERTE NACH DEM,I7,10H-TEN SPIEL,16F6.3)
    RETURN

```

```

C   SUBROUTINE SPIEL(A,N,KX,H,D)
C   PROGRAMM ZUM DURCHFUEHREN EINES SPIELS.
C   WICHTIGE GROESSEN (SOWEIT NICHT IN SUBROUTINE B3 FESTGELEGT)
C   A (J) AUSZAHLUNG DES SPIELS FUER SPIELER J
C   E (J) KARTENSUMME FUER SPIELER J
C   G (J) INDIZIERT EINE AS BEI SPIELER J
C   F (J) INDIZIERT OB SPIELER J NOCH KARTE MOECHTE
C   DIMENSION A(8),KX(16)
C   INTEGER E(8),F(8),G(8),H(8),D(8)
    DO 5J=1,N
      A(J)=0
      F(J)=0
      G(J)=0
      E(J)=0
5    I=IZUF(KX,1)
C   AUSTEILEN DER 1. KARTE
    DO 15 J=1,N
      I=IZUF(KX,0)
      IF(I.EQ.1) G(J)=1
15  E(J)=I+E(J)
C   AUSTEILEN DER 2. KARTE
    DO 20 J=2,N
      I=IZUF(KX,0)
      IF(I.EQ.1) G(J)=1
      E(J)=E(J)+I
C   TEST AUF BLACK JACK
20  IF(E(J).EQ.11.AND.G(J).EQ.1) F(J)=2
      IF(F(J).EQ.2) E(J)=21
C   BEFRAGEN DER SPIELER, OB SIE NOCH KARTEN MOECHTEN
    DO 25 J=2,N
      IF(F(J).EQ.2) GOTO 25
26  CALL SP(J,E,F,G,H,D)

```

```

IF(F(J).EQ.1) GOTO 25
I=IZUF(KX,0)
IF(I.EQ.1) G(J)=1
E(J)=I+E(J)
GOTO 26
25 CONTINUE
C KARTEN FUER CROUPIER
I=IZUF(KX,0)
IF(I.EQ.1) G(1)=1
E(1)=E(1)+I
IF(E(1).EQ.11.AND.G(1).EQ.1) F(1)=2
IF(F(1).EQ.2) E(1)=21
IF(F(1).EQ.2) GOTO 100
27 CALL SP(1,E,F,G,H,D)
IF(F(1).EQ.1) GOTO 100
I=IZUF(KX,0)
IF(I.EQ.1) G(1)=1
E(1)=I+E(1)
GOTO 27
C BERECHNEN DER SPIELAUZAHLUNGEN
100 DO 30 J=2,N
IF(E(J).LE.21) GOTO 35
CALL AUSZ(J,-1.,A)
GOTO 30
35 IF(F(J).GE.2) GOTO 40
IF(E(1).LE.21) GOTO 45
CALL AUSZ(J,1.,A)
GOTO 30
45 IF(E(1)-E(J)) 50,30,55
50 CALL AUSZ(J,1.,A)
GOTO 30
55 CALL AUSZ(J,-1.,A)
GOTO 30
40 IF(F(1).GE.2) GOTO 30
CALL AUSZ(J,1.5,A)
30 CONTINUE
RETURN

SUBROUTINE AUSZ(J,R,A)
C HILFSROUTINE FUER BERECHNEN DER SPIELAUZAHLUNGEN
DIMENSION A(8)
A(J)=R
A(1)=A(1)-R
RETURN

FUNCTION IZUF(KX,KY)
C PROGRAMM ZUM ZUFALLIGEN AUSTEILEN VON 260 KARTEN
C WICHTIGE GROESSEN
C KY=1 INDIZIERT, DASS NEUES SPIEL BEGINNT (ES DARF NEU GEMISCHT WER
C KXXK(J), J=1,13 ANZAHL NOCH NICHT AUSGETEILTER KARTEN DER TYPE J
C KX (14) ANZAHL NOCH NICHT AUSGETEILTER KARTEN, INSGESAMT
C KX (15) LETZTE ERZEUGTE ZUFALLSZAHL
C KX (16) MAXIMALE ANZAHL NOCH NICHT AUSGETEILTER KARTEN EINES TYPES
C NRAND ZUFALLSZAHLENGENERATOR
C DIMENSION KX(16)

```



```

C      IF(KX(14).GT.50.OR.KY.EQ.0) GOTO 5
      MISCHEN
      DO 10 J=1,13
10     KX(J)=20
      KX(14)=260
      KX(16)=20
      GOTO 25
C      ZIEHEN EINER KARTE (NACH GEOMETRISCHER METHODE)
      5     I=KX(15)
      I=NRAND(I)
      Z=I/2.**35.
      I1=1+INT(Z*13)
      KX(15)=NRAND(I)
      Z=KX(15)/2.**35.
      I2=1+INT(Z*KX(16))
      IF(KX(I1)-I2) 5,15,15
15     IZUF=MIN0(I1,10)
      KX(I1)=KX(I1)-1
      KX(14)=KX(14)-1
      KX(16)=0
      DO 20 J=1,13
20     IF(KX(16).LT.KX(J)) KX(16)=KX(J)
      25    CONTINUE
      RETURN

```

```

      SUBROUTINE SP(J,E,F,G,H,D)
C      PROGRAMM ZUR SIMULATION DES SPIELERVERHALTENS
C      WICHTIGE GROESSEN SIEHE HAUPTPROGRAMM
      INTEGER E(8),F(8),G(8),H(8),D(8)
C      ABFRAGE, OB SPIELER AS BESITZT
      IF(G(J).EQ.1) GOTO 100
      GOTO 200
C      ABFRAGE, OB KARTENSUMME UEBER D(J) (MIT AS)
100    IF(E(J)+10.LT.D(J)) GOTO 200
      IF(E(J)-11) 5,5,10
      5     E(J)=E(J)+10
      F(J)=1
      10    G(J)=0
C      ABFRAGE, OB KARTENSUMME UEBER M(J) (OHNE AS)
200    IF(E(J).GT.H(J)) F(J)=1
      RETURN
      END

```

```

ANZAHL SPIELER =2      JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN  H=16  D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN  H=15  D=16
DIE WERTE NACH DEM 5000-TEN SPIEL  .033  .014  -.033  .014
DIE WERTE NACH DEM 10000-TEN SPIEL .032  .010  -.032  .010
DIE WERTE NACH DEM 15000-TEN SPIEL .046  .008  -.046  .008
DIE WERTE NACH DEM 20000-TEN SPIEL .047  .007  -.047  .007
DIE WERTE NACH DEM 25000-TEN SPIEL .045  .006  -.045  .006
DIE WERTE NACH DEM 30000-TEN SPIEL .044  .006  -.044  .006
DIE WERTE NACH DEM 35000-TEN SPIEL .048  .005  -.048  .005
ANZAHL SPIELER =2      JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN  H=16  D=17

```

DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=15 D=17
DIE WERTE NACH DEM 5000-TEN SPIEL .038 .014 -.038 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .038 .010 -.038 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .045 .008 -.045 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .043 .007 -.043 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .043 .006 -.043 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .043 .006 -.043 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .045 .005 -.045 .005
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=15 D=19
DIE WERTE NACH DEM 5000-TEN SPIEL .051 .014 -.051 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .049 .010 -.049 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .044 .008 -.044 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .046 .007 -.046 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .045 .006 -.045 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .043 .006 -.043 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .044 .005 -.044 .005
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=15 D=20
DIE WERTE NACH DEM 5000-TEN SPIEL .048 .014 -.048 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .048 .010 -.048 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .045 .008 -.045 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .050 .007 -.050 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .054 .006 -.054 .006
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=15 D=18
DIE WERTE NACH DEM 5000-TEN SPIEL .059 .014 -.059 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .051 .010 -.051 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .054 .008 -.054 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .048 .007 -.048 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .046 .006 -.046 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .044 .006 -.044 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .045 .005 -.045 .005
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=16 D=19
DIE WERTE NACH DEM 5000-TEN SPIEL .032 .014 -.032 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .046 .010 -.046 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .047 .008 -.047 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .047 .007 -.047 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .048 .006 -.048 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .048 .006 -.048 .006
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=12 D=19
DIE WERTE NACH DEM 5000-TEN SPIEL .026 .014 -.026 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .031 .010 -.031 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .036 .008 -.036 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .041 .007 -.041 .007

DIE WERTE NACH DEM 25000-TEN SPIEL .042 .006 -.042 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .045 .006 -.045 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .044 .005 -.044 .005
DIE WERTE NACH DEM 40000-TEN SPIEL .043 .005 -.043 .005
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=13 D=19
DIE WERTE NACH DEM 5000-TEN SPIEL .033 .014 -.033 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .056 .010 -.056 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .044 .008 -.044 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .040 .007 -.040 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .039 .006 -.039 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .040 .006 -.040 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .044 .005 -.044 .005
DIE WERTE NACH DEM 40000-TEN SPIEL .041 .005 -.041 .005
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=14 D=19
DIE WERTE NACH DEM 5000-TEN SPIEL .010 .014 -.010 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .016 .010 -.016 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .026 .008 -.026 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .031 .007 -.031 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .033 .006 -.033 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .036 .006 -.036 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .038 .005 -.038 .005
DIE WERTE NACH DEM 40000-TEN SPIEL .038 .005 -.038 .005
DIE WERTE NACH DEM 45000-TEN SPIEL .041 .005 -.041 .005
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=14 D=18
DIE WERTE NACH DEM 5000-TEN SPIEL .021 .014 -.021 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .026 .010 -.026 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .030 .008 -.030 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .035 .007 -.035 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .033 .006 -.033 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .034 .006 -.034 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .033 .005 -.033 .005
DIE WERTE NACH DEM 40000-TEN SPIEL .034 .005 -.034 .005
DIE WERTE NACH DEM 45000-TEN SPIEL .035 .005 -.035 .005
DIE WERTE NACH DEM 50000-TEN SPIEL .034 .004 -.034 .004
DIE WERTE NACH DEM 55000-TEN SPIEL .035 .004 -.035 .004
DIE WERTE NACH DEM 60000-TEN SPIEL .035 .004 -.035 .004
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=14 D=19
DIE WERTE NACH DEM 5000-TEN SPIEL .011 .014 -.011 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .029 .010 -.029 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .031 .008 -.031 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .035 .007 -.035 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .036 .006 -.036 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .033 .006 -.033 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .035 .005 -.035 .005
DIE WERTE NACH DEM 40000-TEN SPIEL .034 .005 -.034 .005

DIE WERTE NACH DEM 45000-TEN SPIEL .033 .005 -.033 .005
DIE WERTE NACH DEM 50000-TEN SPIEL .034 .004 -.034 .004
DIE WERTE NACH DEM 55000-TEN SPIEL .034 .004 -.034 .004
DIE WERTE NACH DEM 60000-TEN SPIEL .035 .004 -.035 .004
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=14 D=14
DIE WERTE NACH DEM 5000-TEN SPIEL .064 .014 -.064 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .058 .010 -.058 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .057 .008 -.057 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .054 .007 -.054 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .057 .006 -.057 .006
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=14 D=15
DIE WERTE NACH DEM 5000-TEN SPIEL .039 .014 -.039 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .053 .010 -.053 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .048 .008 -.048 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .054 .007 -.054 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .053 .006 -.053 .006
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=14 D=16
DIE WERTE NACH DEM 5000-TEN SPIEL .048 .014 -.048 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .047 .010 -.047 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .048 .008 -.048 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .045 .007 -.045 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .047 .006 -.047 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .045 .006 -.045 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .048 .005 -.048 .005
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=14 D=17
DIE WERTE NACH DEM 5000-TEN SPIEL .050 .014 -.050 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .036 .010 -.036 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .035 .008 -.035 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .033 .007 -.033 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .031 .006 -.031 .006
DIE WERTE NACH DEM 30000-TEN SPIEL .031 .006 -.031 .006
DIE WERTE NACH DEM 35000-TEN SPIEL .034 .005 -.034 .005
DIE WERTE NACH DEM 40000-TEN SPIEL .036 .005 -.036 .005
DIE WERTE NACH DEM 45000-TEN SPIEL .036 .005 -.036 .005
DIE WERTE NACH DEM 50000-TEN SPIEL .036 .004 -.036 .004
DIE WERTE NACH DEM 55000-TEN SPIEL .036 .004 -.036 .004
ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
ZUGELASSENER RELATIVER FEHLER = .1200
DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=15 D=18
DIE WERTE NACH DEM 5000-TEN SPIEL .052 .014 -.052 .014
DIE WERTE NACH DEM 10000-TEN SPIEL .047 .010 -.047 .010
DIE WERTE NACH DEM 15000-TEN SPIEL .037 .008 -.037 .008
DIE WERTE NACH DEM 20000-TEN SPIEL .042 .007 -.042 .007
DIE WERTE NACH DEM 25000-TEN SPIEL .040 .006 -.040 .006

DIE WERTE NACH DEM 30000-TEN SPIEL .044 .006 -.044 .006
 DIE WERTE NACH DEM 35000-TEN SPIEL .047 .005 -.047 .005
 ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
 ZUGELASSENER RELATIVER FEHLER = .1200
 DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
 DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=12 D=18
 DIE WERTE NACH DEM 5000-TEN SPIEL .020 .014 -.020 .014
 DIE WERTE NACH DEM 10000-TEN SPIEL .027 .010 -.027 .010
 DIE WERTE NACH DEM 15000-TEN SPIEL .027 .008 -.027 .008
 DIE WERTE NACH DEM 20000-TEN SPIEL .032 .007 -.032 .007
 DIE WERTE NACH DEM 25000-TEN SPIEL .035 .006 -.035 .006
 DIE WERTE NACH DEM 30000-TEN SPIEL .040 .006 -.040 .006
 DIE WERTE NACH DEM 35000-TEN SPIEL .037 .005 -.037 .005
 DIE WERTE NACH DEM 40000-TEN SPIEL .038 .005 -.038 .005
 DIE WERTE NACH DEM 45000-TEN SPIEL .039 .005 -.039 .005
 ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
 ZUGELASSENER RELATIVER FEHLER = .1200
 DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
 DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=13 D=18
 DIE WERTE NACH DEM 5000-TEN SPIEL .031 .014 -.031 .014
 DIE WERTE NACH DEM 10000-TEN SPIEL .039 .010 -.039 .010
 DIE WERTE NACH DEM 15000-TEN SPIEL .040 .008 -.040 .008
 DIE WERTE NACH DEM 20000-TEN SPIEL .044 .007 -.044 .007
 DIE WERTE NACH DEM 25000-TEN SPIEL .043 .006 -.043 .006
 DIE WERTE NACH DEM 30000-TEN SPIEL .045 .006 -.045 .006
 DIE WERTE NACH DEM 35000-TEN SPIEL .045 .005 -.045 .005
 ANZAHL SPIELER =2 JEDES 5000-TE SPIEL WIRD AUSGEDRUCKT
 ZUGELASSENER RELATIVER FEHLER = .1200
 DIE VERHALTENSPARAMETER FUR SPIELER 1 LAUTEN H=16 D=17
 DIE VERHALTENSPARAMETER FUR SPIELER 2 LAUTEN H=11 D=18
 DIE WERTE NACH DEM 5000-TEN SPIEL .062 .014 -.062 .014
 DIE WERTE NACH DEM 10000-TEN SPIEL .059 .010 -.059 .010
 DIE WERTE NACH DEM 15000-TEN SPIEL .051 .008 -.051 .008
 DIE WERTE NACH DEM 20000-TEN SPIEL .054 .007 -.054 .007
 DIE WERTE NACH DEM 25000-TEN SPIEL .055 .006 -.055 .006
 DIE OPTIMALSTRATEGIE LAUTET H=14. D=18.
 MIT EINER VERLUSTWAHRSCHEINLICHKEIT VON $P = -.03517$

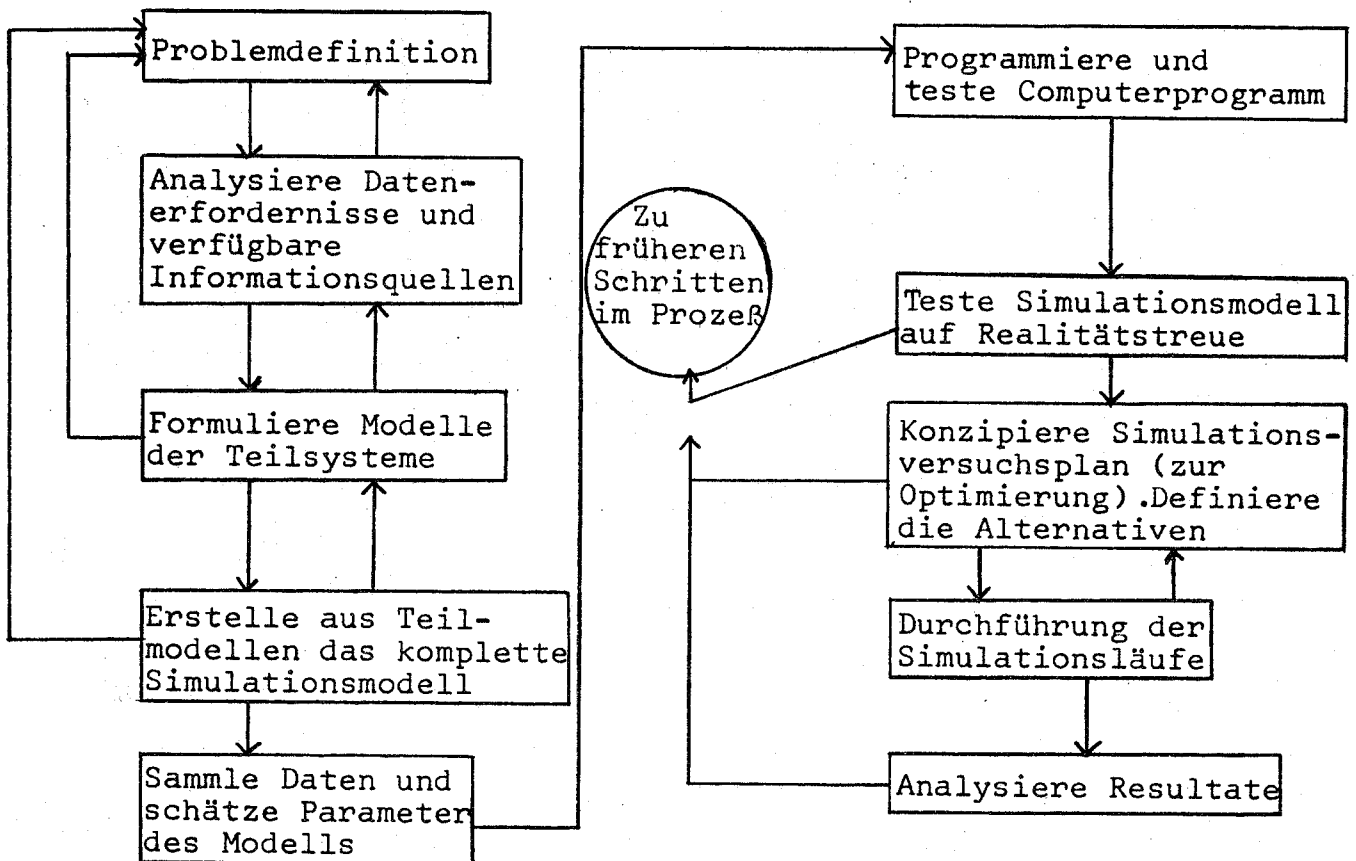
5. METHODIK BEIM ERSTELLEN EINES SIMULATIONSMODELLS

5.1. Einleitung

In diesem Kapitel wollen wir das Vorgehen bei der Entwicklung eines Simulationsmodells nochmals genauer beschreiben und die Erkenntnisse, welche aus den letzten Kapiteln diesbezüglich gewonnen wurden, zusammenzufassen. Ziel ist nun, die einzelnen Schritte beim Aufbau eines Simulationsmodells zu identifizieren und zu beschreiben. Natürlich hängt die Reihenfolge dieser Schritte vom speziellen Problem ab, mitunter werden auch nicht alle Schritte vonnöten sein.

Figur 5-1 ist ein Flußdiagramm für die Entwicklung eines Simulationsmodells. Bemerkenswert dabei ist, daß an einigen Stellen der Prozeß wieder zu früheren Schritten zurückkehren kann. Das zeigt einen wichtigen Aspekt beim Modellaufbau. Durch größere Einsicht in die analysierte Wirklichkeit muß immer wieder das Modell angepaßt werden. So kann z.B. eine genauere Analyse der Daten eine Änderung der ursprünglichen Problemstellung bewirken. Bei komplizierten Systemen wird es auch oft passieren, daß das ursprüngliche Modell zu einfach konzipiert war, um die Realität annähernd zu beschreiben oder zu kompliziert konzipiert war, um das Modell innerhalb vernünftiger Rechenzeiten am Computer durchzuspielen.

Modellbau ist mindestens sosehr eine Frage der Intuition, wie der Methodik. Deshalb täuscht das Flußdiagramm eine Exaktheit vor, die eigentlich nicht gegeben ist. Insofern sind die folgenden Erläuterungen bestenfalls Hilfen und ersetzen nicht persönlichen Einfallsreichtum bei speziellen Problemen.



Figur 5-1
Flußdiagramm zur Erstellung eines
Simulationsmodells

5.2. Problemdefinition

Problemdefinierung ist wohl der kritische Schritt beim Aufbau eines Modells. Deswegen ist auch im Flußdiagramm eine oftmalige Rücksprungmöglichkeit zu diesem Schritt gegeben.

Ein erster Schritt bei der Problemformulierung ist die Spezifizierung derjenigen Ziele, welche durch das Simulationsmodell erreicht werden sollen. Zunächst sind diese Ziele in einer qualitativen Form gegeben und müssen erst in operationelle Größen umgesetzt werden. Die Ziele legen aber auch schon weitgehend das Modell fest, da ja dieses im Hinblick auf die Fragen, welche mittels dem Modell angegangen werden sollen, konzipiert wird. Bei Black Jack hätten wir, wäre es uns um das Verhalten der Spieler während des Spielverlaufs gegangen ein völlig anderes Modell erstellen müssen, als wir es für unsere Problemstellung tatsächlich getan haben.

Bei mehrfacher eventuell sogar einander widersprechender Zielsetzung muß diese evtl. mit Hilfe der Entscheidungsanalyse auf eine einzige reduziert werden.

Handelt es sich um ein Simulationsmodell, welches für ein Optimierungsproblem herangezogen werden soll, so müssen nun auch die kontrollierbaren Variablen, also die von außerhalb des Modells steuerbaren Größen, festgelegt werden. Sind wir nun soweit, so können wir daran gehen, unseren Bedarf an Daten sowie deren Verfügbarkeit genauer zu betrachten.

5.3. Datenerfordernisse- und beschaffung

Dieser Abschnitt ist neu, da für das Simulationsmodell Black Jack keine Probleme in diesem Bereich bestanden, weil mit den Spielregeln sämtliche notwendige Information zu unserer Verfügung stand.

Beim Aufbau eines Simulationsmodells werden Daten benötigt für

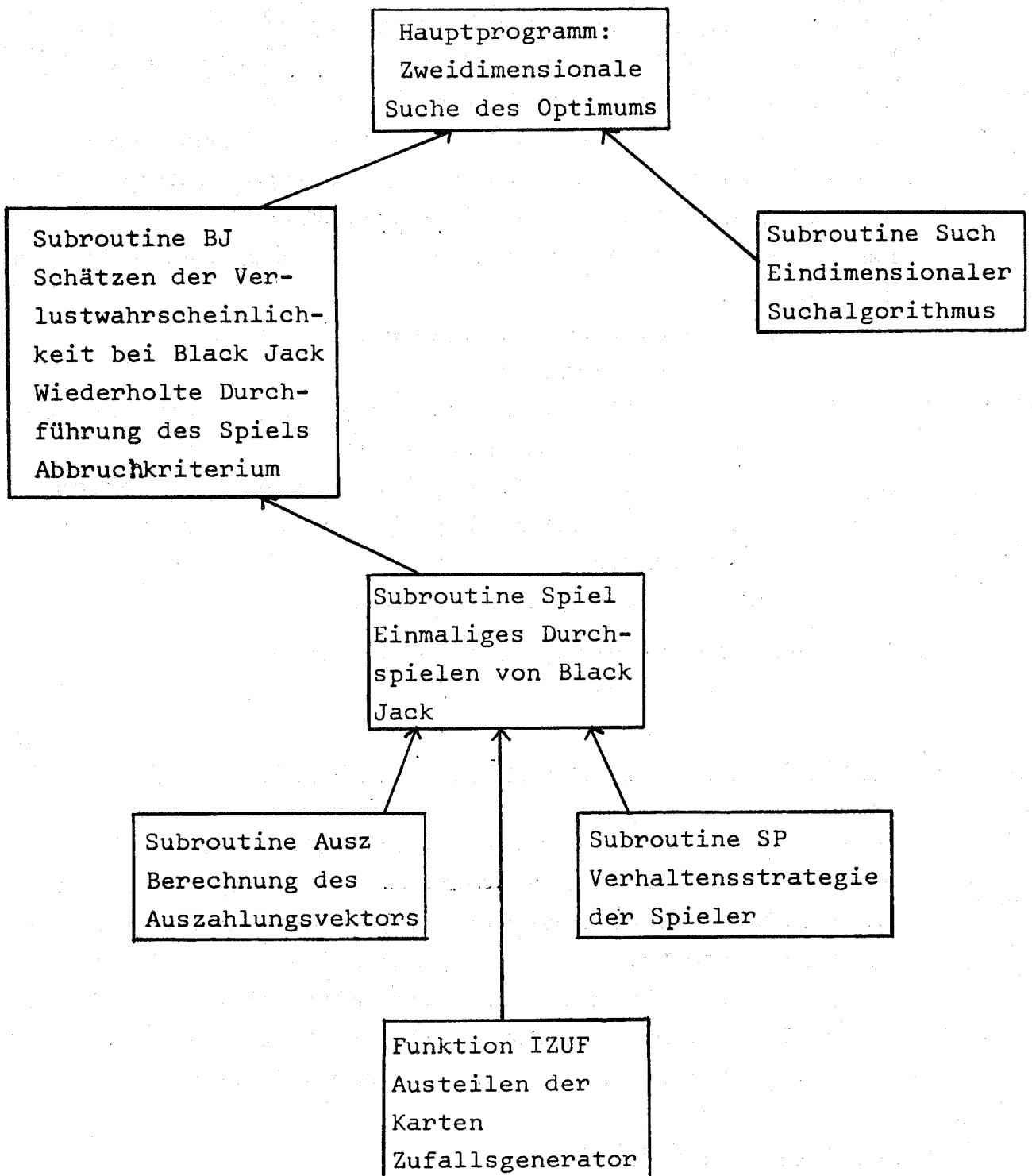
- Schätzung von Konstanten und Parametern
- Bestimmung der Anfangswerte für alle Variablen
- Testen des Modells auf Übereinstimmung mit der Wirklichkeit (Validierung).

Datenbeschaffung ist grundsätzlich kein Problem, kritisch wird es dadurch, daß die benötigten Daten mit wenig Kosten und innerhalb enger Zeitgrenzen erhalten werden sollen. So könnte man ja, falls für das Modell im Moment tatsächlich keine Daten vorhanden sind, im Verlaufe der kommenden Jahre die Daten sammeln, aber diese Möglichkeit besteht kaum jemals. Am einfachsten gestaltet sich die Datenbeschaffung natürlich, falls für den untersuchten gesellschaftlichen Prozeß bereits seit geraumer Zeit Datenaufzeichnung mit Hilfe von EDV durchgeführt wird. Ist dies nicht der Fall, so muß man meist mit Stichproben auskommen oder sogar mit persönlichen Schätzungen von Personen welche mit der Materie besonders vertraut sind. Allerdings muß man auch beim Modell-aufbau berücksichtigen, daß unsichere, vage Daten kein sehr feines, hochstrukturiertes Modell rechtfertigen, welches z.B. für viele Faktoren Resultate liefert, deren Anfangswerte nicht einmal genau bekannt waren. Deshalb besteht eine starke Wechselwirkung zwischen Daten und Modell. Man kann und soll sinnvollerweise kein Modell konstruieren, welches Daten benötigt, die unter den Restriktionen Zeit und Geld nicht beschafft werden können, sondern muß sich mit einem bescheidenen Modell begnügen. Dies gilt natürlich nicht bei der Erstellung eines Modells, bei dem primär die zugrunde liegende Struktur unter verschiedenen Annahmen analysiert werden und nicht auf spezielle, der Realität angepaßten Variablenwerte eingegangen werden soll. Beispiel wäre die Untersuchung von Stabilitätsverhalten bestimmter Modelle.

Klar ist, daß vor allem bei der Datenbeschaffung sehr extensiver Gebrauch von statistischen Methoden gemacht werden muß. Schließlich ist eine korrekte Bestimmung der Modellparameter sowie der Startbedingungen bei der Simulation auch Voraussetzung für die Wirklichkeitstreue des Modells.

5.4. Formulierung der Subsysteme des Modells

Der Aufbau eines Modells wird stark vereinfacht, gelingt es, das Gesamtmodell in Teilmodelle zu zerlegen. Teilmodelle haben vor allem die Eigenschaft, ein gewisses Eigenleben unabhängig vom Gesamtmodell aufzuweisen, wiewohl natürlich über gewisse Variablen mit dem Gesamtmodell verbunden. Programmiertechnisch würde das in FORTRAN bedeuten, daß das Gesamtprogramm in mehrere Subroutinen zerlegt wird, die dann wieder über das Hauptprogramm verkoppelt werden. Nicht nur ist eine solche Konzeption vom Modellbauer in ihrer Struktur leichter überschaubar, auch das Austesten des Programms gestaltet sich so entschieden einfacher. Natürlich sollten auch die Teilmodelle, soweit möglich, in weitere Teile aufgespalten werden, sodaß man schließlich zu einem hierarchischen Aufbau eines Simulationsmodells gelangt. Figur 5-2 zeigt als Beispiel den hierarchischen Aufbau des Black-Jack Modells.



Figur 5-2

5.5. Erstellung des Simulationsmodells aus den Teilsystemen

Nun wird es Zeit, die Teilsysteme zu programmieren und zum Gesamtmodell zusammenzufügen. Vorher muß aber noch festgelegt werden in welcher Programmiersprache das Modell geschrieben werden soll. Wir haben für Black Jack FORTRAN gewählt, aber bei komplexeren, vor allem diskreten Simulationsmodellen eignen sich andere Sprachen wohl besser. Die Entscheidung sollte vor allem folgende Kriterien einkalkulieren

- Effizienz verschiedener Computersprachen bei der Compilierung und Rechenzeit
- Schwierigkeit des Übersetzens der (verbal) formulierten Teilsysteme in die Computersprache
- Der gewünschte Output der Simulation und den Programmieraufwand in der entsprechenden Computersprache, um den Output zu erhalten.
- Die Kenntnisse des Programmierers.

Auf die verschiedenen Programmiersprachen werden wir später noch genauer eingehen.

5.6. Schätzen der Simulationsvariablen- und parametern

Gemäß den Ausführungen in Abschnitt 5.3. müssen nun die relevanten Größen geschätzt werden. Je nachdem, ob es sich um korrelierte oder unkorrelierte, zeitlich veränderliche oder konstante Größen handelt, bereitet dies mehr oder weniger Aufwand. Wir wollen jedoch nicht näher darauf eingehen. Ein sauberer Simulationsmodell-Aufbau ist aber ohne statistische Kenntnisse wohl kaum durchführbar.

5.7. Austesten des Modells

Ist das Modell nun programmiert, so muß es bzw. dessen Teilsysteme auf formale und logische Richtigkeit überprüft werden. Vor allem hier bewährt es sich, hat man kleine, überschaubare Teilsysteme programmiert, welche man unabhängig voneinander testen kann.

5.8. Validierung des Modells

Bevor man das Modell für die ursprüngliche Problemstellung benutzen kann, muß man sich vergewissern, daß es gelungen ist, die Wirklichkeit adäquat im Modell zu erfassen. Man muß also die Resultate von Simulationsläufen mit der Realität vergleichen. Dies ist bei Prognosenmodellen möglich, in dem man mittels des Modells bereits eingetretene, also historische Entwicklungen aus noch früheren zu prognostizieren versucht. Die Güte dieser Prognose ermöglicht dann Aussagen über die Güte des Modells. Welche Abweichung man toleriert liegt dabei im Ermessen des Beurteilers.

5.9. Optimierung und Analyse der Resultate, Sensitivitätsanalyse

Mit Optimierung wollen wir uns nicht weiter beschäftigen, das haben wir bereits ausführlich getan und es wird hier nur der Vollständigkeit halber aufgeführt.

Die Analyse der Resultate bzw. deren Interpretation ist der Schluß im Aufbau eines Simulationsmodells. Erhält man Antworten, welche im Widerspruch zu den intuitiven Schlußfolgerungen stehen, so sollte man jedenfalls versuchen, diesen Widerspruch zu klären. Bei statistischen Resultaten muß man sich über Simulationsdauer bzw. Genauigkeit der statistischen Schätzungen klar werden, aber auch dabei nicht exaktere Resultate fordern, als man Daten in das Modell eingespeist hat.

Um die Verlässlichkeit der erhaltenen Resultate abschätzen zu können, ist eine Sensitivitätsanalyse erforderlich. Gemeint ist das Variieren von Anfangswerten oder Modellkonstanten um kleine Beträge und der Vergleich der so erhaltenen Ergebnisse mit den ursprünglichen. Reagiert das Modell auf bestimmte Veränderungen sehr sensibel, also mit stark abweichenden Resultaten, so muß man die kritischen Größen besonders exakt bestimmen oder bei dem Resultat sehr

stark streuende Werte in Kauf nehmen. Sensitivitätsanalysen in vielen Größen ist, wie Optimierung, ein aufwendiges Unterfangen, zur Beurteilung der Güte der Resultate aber unerlässlich.

So wäre beim Black Jack Modell eigentlich zu untersuchen, ob ein kleiner relativer Fehler bei der Schätzung der Versuchswahrscheinlichkeit, welcher ja mit $\epsilon = K \cdot |\bar{X}|$ festgelegt wurde, eine große Änderung bei der Optimalstrategie bewirkt. Man müßte also das Modell nochmals mit einem anderen K-Wert durchrechnen.

Kein Resultat eines Simulationsmodells darf blind akzeptiert werden. Immer müssen mit Hilfe der persönlichen Einsicht Vereinfachung bei der Modellerstellung, Unsicherheit der Daten und ähnliches bei der Interpretation mitberücksichtigt werden.

6.

AMBULANZSYSTEM VON ZÜRICH

6.1. Einleitung

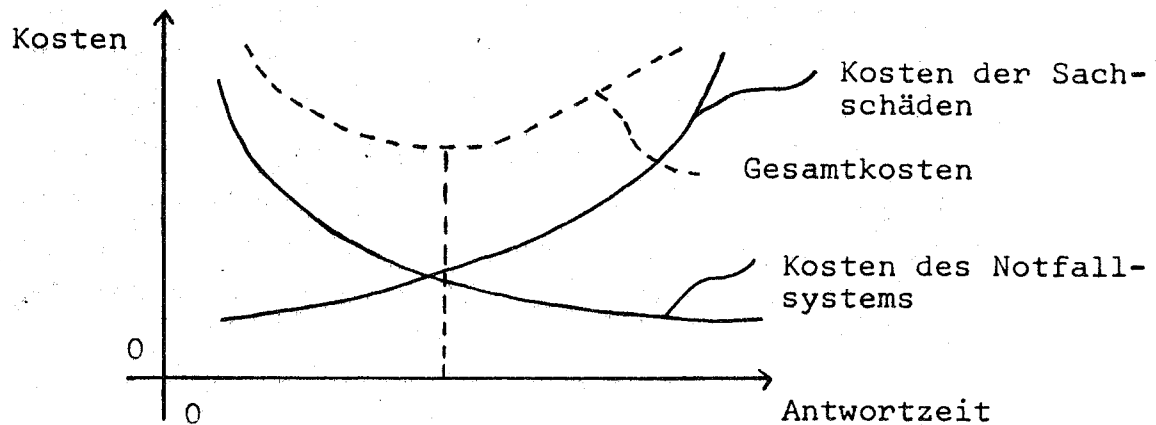
Als erste praktische Anwendung soll nun der organisatorische Ablauf des Rettungsdienstes von Zürich simuliert werden. Diese Studie wurde zunächst als Übungsbeispiel in der Vorlesung von Th.M.Liebling, ETH-Zürich, behandelt und daran anschließend als Semesterarbeit von J. Polymeris und J.Riera. Wir können hier nur ein vereinfachtes Modell, wie es in etwa als Übungsbeispiel verwendet wurde, analysieren. Hierbei handelt es sich um ein diskretes, stochastisches und stationäres Modell, wie es typisch ist für eine betriebswirtschaftliche Anwendung. In diesem Kapitel wollen wir die Aufgabe genau beschreiben, welche Daten benötigt werden, sodaß wir uns im darauffolgenden Kapitel mit der Programmierung befassen können.

6.2. Notfallssysteme

Das betrachtete System ist ein spezieller Fall der allgemeinen Klasse von Notfallssystemen. Darunter versteht man kommunale Einrichtungen, deren Aufgabe es ist, bei unvorhersehbaren Ereignissen, welche eine Bedrohung von Leben oder Sachwerten beinhalten, möglichst rasch helfend einzugreifen. Beispiele solcher Notfallssysteme sind neben Rettungsdiensten auch Feuerwehr, Polizei aber auch weniger bekannte Dienste, wie solche zur Behebung von Gas- oder Wasserrohrbrüchen. Allen diesen Organisationen ist gemeinsam, daß die zur Hilfe nötigen Einheiten (wie Fahrzeuge, Personal, Geräte etc.) an meistens festen Punkten innerhalb der zu bedienenden Region stationiert sind, daß die unvorhersehbaren Ereignisse zu zufälligen Zeitpunkten üblicherweise durch einen telephonischen Anruf angezeigt werden und daß dann eine Hilfseinheit, wie Rettungswagen plus Personal oder Feuerwehrauto plus Personal, zum Ereignisort fahren soll. Die Qualität eines Notfallsystems zeigt sich also insbesondere daran, wie schnell eine Hilfseinheit auf Grund eines Telefonanrufs beim Ereignisort sein kann. Natürlich kommt es auch auf die Qualität der Hilfs-

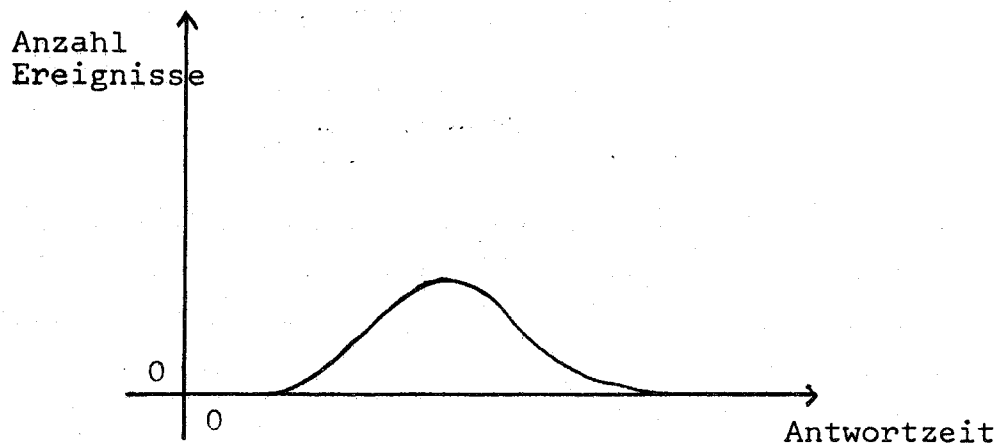
einheit in Bezug auf deren Personal und Geräteausstattung an, auf diesen Punkt wollen wir aber hier nicht näher eingehen. Die Antwortzeit (das ist die Zeitdifferenz zwischen Eintreffen des Hilfspersonals beim Ereignisort und dem Telefonanruf, auf dem dieser Einsatz beruht) hängt primär ab von der Distanz zwischen Ereignisort und dem hinbeordneten Einsatzfahrzeug. Nimmt man an, wie es vielleicht mit Ausnahme von Polizeieinsätzen meistens der Fall ist, daß die Hilfseinheiten an mehreren Einsatzstellen fest stationiert sind, so hängt also vor allem von der Lage dieser Einsatzstellen die Antwortzeit ab. Weiters ist aber die Anzahl Hilfseinheiten, welche an den einzelnen Einsatzstellen stationiert sind von Bedeutung, da ja, falls sehr wenige solcher Einheiten vorhanden sind, durchaus der Fall eintreten kann, daß bei Eintreffen eines Notfalleinrufes alle Hilfseinheiten gerade bei einem anderen Notfallereignis beschäftigt sind und sich somit die Antwortzeit um die Wartezeit auf eine freie Hilfseinheit verlängert. Somit sind die wesentlichen Entscheidungsparameter beim Planen eines Notfallsystems erstens die Anzahl und der Ort von Einsatzstellen und zweitens die Anzahl Hilfseinheiten, die an jeder Einsatzstelle stationiert sind. Diese kann zusätzlich variabel sein in Bezug auf Tageszeit, Tag oder Monat, um diesbezügliche Schwankungen bei der Häufigkeit von Notfallereignissen besser zu berücksichtigen. All diese Entscheidungsparameter sind natürlich unter der Restriktion zu bestimmen, daß nur eine beschränkte finanzielle Unterstützung durch die kommunale Verwaltung gegeben wird, was darauf hinausläuft, daß die Anzahl Hilfseinheiten und Einsatzstellen eine bestimmte Schranke nicht übersteigen darf. Man muß sich dabei vor Augen halten, daß die genannte finanzielle Restriktion implizit eine ökonomische Bewertung von Menschenleben beinhaltet, da durch Limitierung natürlich die Qualität eines Notfalldienstes, also die Antwortzeit, limitiert ist, wodurch evtl. Menschenleben, welche durch eine raschere Antwortzeit gerettet werden könnten, preisgegeben werden. Einfacher ist es bei Notfallsystemen primär zur

Vermeidung von Sachschäden, wie z.B. Feuerwehr. Hier könnte man die Kosten des Notfallssystems gegenüber den Kosten der Sachschäden abwägen und gemäß Figur 6-1 die kostengünstigste Lösung bestimmen.



Figur 6-1

Bei einem gegebenen Notfallsystem wird die Antwortzeit nicht für alle Ereignisse diegleiche sein, sondern eher eine Verteilung gemäß Figur 6-2 aufweisen.

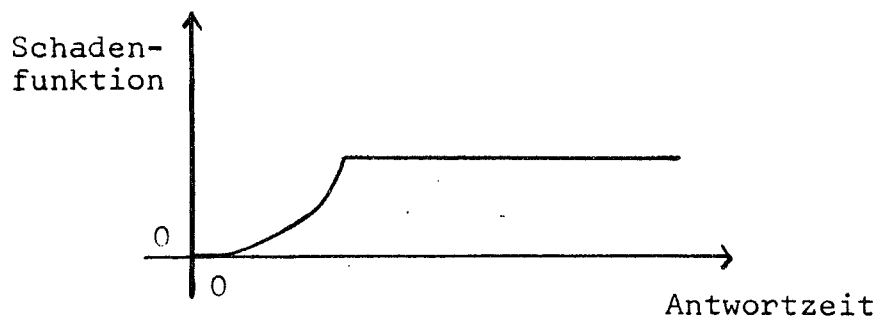


Figur 6-2

Dann entsteht die Frage, wie denn die Qualität zweier verschiedener Notfallsysteme mit zwei verschiedenen Antwortzeitverteilungen verglichen werden kann. Das naheliegendste Vergleichskriterium stellt der Erwartungswert der Verteilung dar, aber nicht immer ist dies sinnvoll, da jede Antwortzeit gleich bewertet wird und höhere Antwortzeiten evtl. stärker gewichtet werden sollten, weil sie gefährlicher sind. Dies bedeutet, daß über die Antwortzeiten t eine Schadenfunktion $u(t)$ definiert wird, etwa gemäß Figur 6-3 und dann die Antwortzeitverteilung $p(t)$ mit der Schadenfunktion $u(t)$ gewichtet in der Form

$$\int_0^{\infty} p(t)u(t)dt \quad (6-1)$$

als Vergleichskriterium gewählt wird (je kleiner 6-1 ist, desto besser).



Figur 6-3.

Daß $u(t)$ für große t konstant wird ist deshalb einleuchtend, da über einen kritischen Zeitbereich hinaus der Einsatz einer Hilfskraft ohne Bedeutung ist, da dann der betroffene Mensch tot, das Haus abgebrannt ist, also keine nennenswerte Hilfe mehr geleistet werden kann.

Es soll nun speziell das Rettungssystem von Zürich und daran anschließend das vereinfachte Modell, mit dem wir uns beschäftigen werden, beschrieben werden.

6.3. Rettungssystem Zürich

Das Züricher Ambulanzsystem erfüllt neben dem Einsatz bei Notfällen noch zusätzlich den Transport von bettlägerigen Kranken. Diese Transporte sind im voraus bekannt, können nötigenfalls, also bei einem Notfalleinsatz, auch verschoben werden. Derzeit wird die Stadt Zürich von zwei Einsatzstellen aus bedient, wobei, je nach Tageszeit, an einer Einsatzstelle maximal 6 bzw. 2 und minimal (zwischen 0^h - 6^h) 2 bzw. 0 Einheiten stationiert sind. Die Rettungswagen überführen die Patienten in das nächste von fünf zur Wahl stehenden Spitälern im Züricher Stadtgebiet und kehren dann, sofern sie nicht per Funk umdirigiert werden oder einen Krankentransport zu erledigen haben, zur Einsatzstelle zurück.

Da die maximale Anzahl Einheiten, nämlich 8, gegeben war, bestand das organisatorische Problem des Rettungsdienstes in der Frage, ob durch Einführung einer dritten Einsatzstelle an einem vorgegebenen Ort und eine Neuaufteilung der Einheiten auf die drei Einsatzstellen die Qualität des Rettungsdienstes nennenswert verbessert werden kann. Dieses Problem wurde in der Arbeit von J. Polymeris und J. Riera untersucht.

Für Vorlesungszwecke muß das ursprüngliche Problem doch recht einschneidend verändert werden, da zur Simulation des echten Ambulanzsystems ca. 2000 Fortran Statements erforderlich waren, was den Rahmen der Vorlesung wohl sprengen würde.

Der wohl logisch aufwendigste Bereich des Simulationsmodells ist die Vielfalt der Standorte von Einsatzstellen bzw. Spitälern. Wir nehmen daher für unser Modell an, daß ganz Zürich von einer Einsatzstelle aus bedient wird und daß nur ein Spital - am gleichen Ort, wie die Einsatzstelle - existiert. Dann soll die Aufgabe sein, die Qualität des Systems in Bezug auf verschiedene Anzahl Hilfseinheiten, also Rettungswagen, zu bestimmen. Weiters sollen Fahrzeuge,

welche bereits unterwegs sind, nicht mehr umdirigiert werden können und die Belastung des Systems unabhängig von Tageszeit, Tag oder Monat sein. Dieses Modell könnte man dann als Baustein für ein kompliziertes betrachten, falls bei Existenz mehrerer Einsatzstellen diesen feste Gebiete zur Bedienung zugeteilt sind, denn dann könnte man jedes Gebiet mit einer Einsatzstelle unabhängig von der anderen simulieren. Tatsächlich wird in Zürich ein Großteil der Einsätze auf diese Weise gesteuert, also eine Einsatzstelle ist für ein bestimmtes Gebiet zuständig.

6.4. Datenanalyse

Nachdem nun die gestellte Aufgabe soweit spezifiziert ist, gilt es, die benötigten Daten festzulegen und zu beschaffen. Erforderlich ist wohl die Verteilung der Telefonanrufe zur Anforderung eines Rettungswagen für einen Notfall bzw. Kranken. Unter der Annahme tageszeitunabhängiger Anrufintensität ergaben die Daten, daß pro Tag (= 1440 Minuten) 14 Notfälle und 100 Kranke zu bedienen waren und daß die Intervalle zwischen zwei Notfällen bzw. zwei Kranken exponentialverteilt waren. Ein zweites wichtiges Kriterium ist die regionale Verteilung. Hierzu mußte Zürich in kleinere Gebiete zerlegt werden und für jedes Gebiet die Anzahl Notfälle bzw. Kranke pro Jahr akkumuliert werden und daraus eine regionale Wahrscheinlichkeitsverteilung gewonnen werden. Da diese Daten aber nur schwer zugänglich waren, wurde die Anrufshäufigkeit proportional zur entsprechenden Einwohnerdichte angenommen. Dies ergab für 12 Regionen

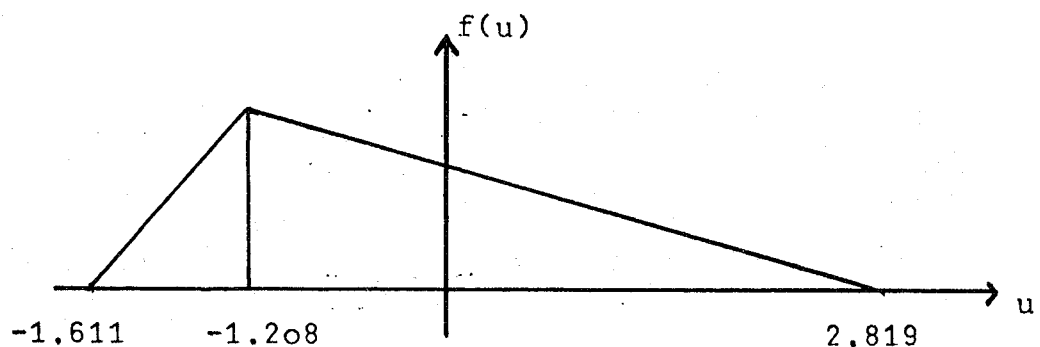
Region	1	2	3	4	5	6	7	8	9	10	11	12
Einwohner- zahl (in Tausend)	9	32.6	51.7	31.8	12.5	37.8	39.1	20.3	47.2	35.8	56.9	33

Beim Erfassen dieser Daten wird angenommen, daß die zeitliche und räumliche Verteilung von Telefonanrufen voneinander unabhängig sind und daß kein Anruf mehr als eine Notfalleinheit anfordert.

Zur Simulation werden weiters die Fahrzeiten von der Einsatzstelle zu den Einsatzorten, also den 12 Regionen, benötigt. Hierbei ist zu unterscheiden zwischen Fahrten mit Blaulicht, bei Notfällen, und solchen ohne, bei Kranken. Da zu einzelnen Punkten in jeder Region die Fahrzeiten unterschiedliche sind, aber auch die Verkehrsverhältnisse wechselnd, müssen Fahrzeiten als stochastische Variablen betrachtet werden. Als erste Näherung kann angenommen werden, daß für die Fahrzeit X_i in die i -te Region bzw. von der i -ten Region zurück gilt

$$X_i = a_i U + b_i \quad (6-2)$$

Hierbei ist U eine Zufallsvariable mit der Dichte $f(u)$, gemäß Figur 6-4.



Figur 6-4

Dann läßt sich zeigen, daß

$$\begin{aligned} E(U) &= 0 \\ \text{Var}(U) &= 1 \end{aligned} \quad (6-3)$$

Daraus folgt aber, daß

$$\begin{aligned} E(X_i) &= b_i \\ \text{Var}(X_i) &= a_i^2, \end{aligned} \quad (6-4)$$

weiters sei die Varianz abhängig von der mittleren Fahrzeit, sodaß

$$a_i = \frac{b_i}{4}. \quad (6-5)$$

Die mittleren Fahrzeiten ergeben sich aus Figur 6-5.

Region	1	2	3	4	5	6	7	8	9	10	11	12
mit Blau- licht	10	15	12	11	11	10.5	13.5	12.5	15	14	14.5	15
ohne Blau- licht (in Minuten)	14	21	17	15	15	14	18.5	17	22	20	21.5	21

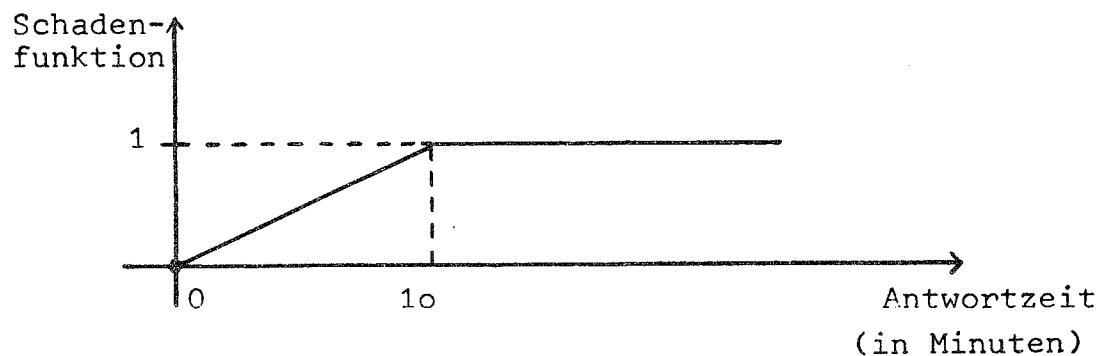
Figur 6-5

Aus 6-5, 6-2 und Figur 6-5 folgt natürlich, daß stets

$$X_i > 0 \quad (6-6)$$

gilt - eine einfache Voraussetzung, die für Fahrzeiten erfüllt sein muß.

Bereits im vorhergehenden Kapitel wurde das Problem der Bewertung der Resultate behandelt. Das Simulationsprogramm soll also die empirische Verteilungsfunktion der Antwortzeit in Abhängigkeit von der Anzahl Rettungswagen bestimmen. Da aber der Vergleich zweier Verteilungsfunktionen schwer möglich ist, soll über die Antwortzeit auch eine Schadenfunktion gemäß Figur 6-6 definiert werden, mit der die Verteilung bewertet werden soll.



Figur 6-6

6.5. Literatur

J.Polymeris & J. Riera, Simulation des Ambulanzdienstes der Stadt Zürich, Studienberichte Nr.1, IFOR, ETH-Zürich, 1974

J.M.Chaiken & R.C.Larson, Methods for Allocating Urban Emergency Units: A Survey, Management Science, Vol.19 (1972), Nr.4, Part 2, pp. P 110-P 130.

6.6. Aufgaben

- 1) Man schreibe Fortran-Subroutinen zur Erzeugung aller für das Ambulanzmodell nötigen Zufallszahlen, wie sie in 6.4.angeführt sind.
- 2) Man entwickle ein Flußdiagramm des Ambulanzmodells.

7.

SIMULATIONSSPRACHEN

7.1. Einleitung

In diesem Kapitel sollen einige Simulationssprachen etwas genauer charakterisiert werden, um die Entscheidung, in welcher Sprache das Modell programmiert werden soll, zu erleichtern. Ein ganzes Kapitel soll den Simulationssprachen gewidmet sein, da es hier wohl um eine der zentralen Fragen der Simulation geht, denn die richtige Wahl kann Programmier- und Rechenzeit wesentlich herabsetzen.

Bevor wir die Vor- und Nachteile der einzelnen Sprachen erörtern, soll gezeigt werden, welche Aufgaben Simulationssprachen i.a. erfüllen sollen.

7.2. Simulationssprachen und -modelle

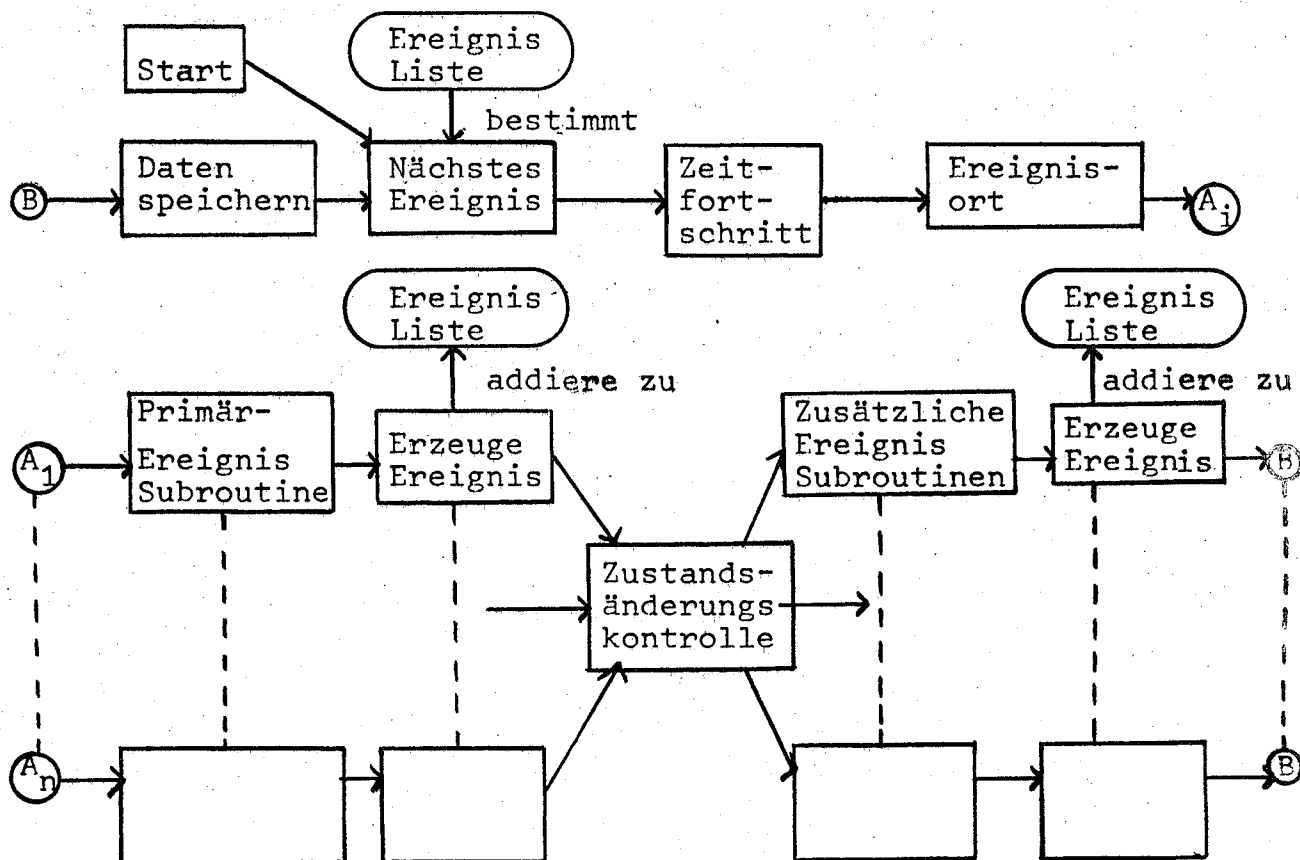
Betrachtet man den Vorgang zum Erstellen eines Simulationsprogramms, so erfolgt die erste Beschreibung des Modells in der Umgangssprache. Diese Beschreibung muß nun in eine dem Computer verständliche Sprache transferiert werden. Ist diese Sprache FORTRAN, so besteht doch eine erhebliche Differenz zwischen den beiden Formulierungen und der Übersetzungsprozeß ist aufwendig. Außerdem zeigt sich, daß Simulationsprogramme gewisse Strukturen gemeinsam haben, die immer aufs Neue zu programmieren eigentlich unnütz ist. Fast alle Simulationsprogramme weisen die folgenden gemeinsamen Charakteristika auf:

- Erzeugen von gleichverteilten Zufallszahlen
- Erzeugen von Zufallszahlen spezieller Verteilungen
- Zeitfortschritt, entweder um eine Einheit oder zum nächsten Ereignis
- Datenspeicherung für Output
- Statistische Analyse gespeicherter Daten
- Spezielle Outputformate
- Aufspüren logischer Inkonsistenzen und anderer Fehler.

Simulationssprachen haben daher primär den Vorteil, daß für die erwähnten Anforderungen bereits spezielle Subroutinen vorhanden sind und nicht mehr programmiert werden müssen. Ein gravierender Unterschied besteht noch zwischen den Arten des Zeitfortschritts und ob es sich um den Fluß ganzzahliger Elemente durch einen Prozeß handelt. Die hierfür entwickelten diskreten Simulationssprachen haben meistens noch zusätzliche Möglichkeiten:

- Bestimmt Ereignisart (nach Abruf aus einer Ereignisliste)
- Ruft Subroutinen, um die Zustandsvariablen auf Grund des Ereignisses zu verändern
- Identifiziert spezielle Zustandsart
- Speichert und ruft Daten von Listen (Tabellen oder Felder) ab.

Die Struktur eines diskreten Simulationsmodells zeigt Figur 7-1.



Figur 7-1

Das Wichtigste bei diskreten Simulationsmodellen ist die Darstellungsart des Prozesses. Ein Prozeß ist eine Aktivität, die in der Zeit abläuft. Beginn, Änderungen und Ende des Prozesses heißen Ereignisse. Das System ändert seinen Zustand genau dann, wenn ein Ereignis eintritt. In einer diskreten Simulation wird nun der Prozeß nicht explizit modelliert, sondern die Ereignisse, welche den Zustand des Prozeßablaufes verändern. Änderungen sind dabei stets plötzliche Veränderungen des Zustands und nicht solche, die sich kontinuierlich mit der Zeit verändern.

Die erste Aufgabe bei der Simulation ist die Erstellung der Ereignisliste, welche alle möglichen Ereignisse enthält. Im Ablauf des Programms enthält dann die Ereignisliste stets die aktuellen Zeitpunkte zu denen künftige Ereignisse passieren, wird dabei natürlich durch eintretende Ereignisse laufend verändert.

Die Ereignis-Zeit Vorhersage ist der zentrale Teil eines diskreten Simulationsmodells, enthält sie doch des Modellbauers Hypothese, wie das System und die Umwelt den Prozeßverlauf bestimmen. In manchen Fällen wird die Ereignis-Zeit deterministisch in anderen stochastisch festgelegt sein.

Gerade aus Figur 7-1 kann man erkennen, welche Vorteile eine Simulationssprache bringen kann, indem insbesondere das Updaten der Ereignisliste und das Festlegen des nächsten Ereignisses einfacher programmiert werden kann, als in einer allgemeinen Programmiersprache, wie etwa FORTRAN.

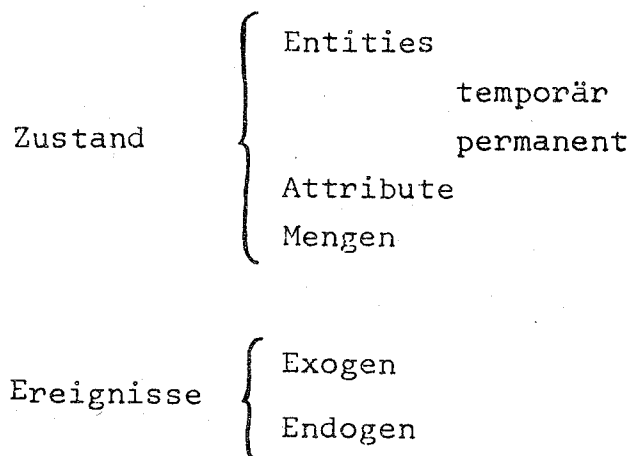
Der Hauptgewinn bei Verwendung einer Simulationssprache ist die Einsparung an Zeit zum Programmieren und Testen des Simulationsprogramms, natürlich unter der Annahme, daß dem Modellbauer FORTRAN und die gewählte Simulationssprache gleich geläufig sind. Dazu kommt, daß ein Modell geschrieben in einer Simulationssprache einfacher zu modifizieren ist, als eines in FORTRAN.

Neben diesen Vorteilen ist aber eines nicht zu vergessen: Simulationssprachen sind in gewissem Sinn echte Sprachen, d.h. sie eignen sich zur Beschreibung eines Prozesses ungeachtet dessen, daß sie auch von einem Computer verarbeitet werden können. Und, wie mit anderen Sprachen, beginnt man beim Erstellen eines Simulationsmodells in der Terminologie der Simulationssprache zu denken. Daher kann die Simulationssprache selbst eine Hilfe bei der Modellformulierung sein.

7.3. Ein Beispiel einer Simulationssprache für diskrete Simulationen: SIMSCRIPT

Die, neben GPSS (General Purpose Simulation System), wohl verbreitetste Simulationssprache für diskrete Simulationen ist SIMSCRIPT. Es soll nicht Aufgabe des Skriptums sein, eine detaillierte Einführung in SIMSCRIPT zu bieten, indes soll die Struktur der Sprache im Folgenden beschrieben werden und zum Schluß sowohl eine FORTRAN wie eine SIMSCRIPT-Version des Ambulanzmodells dargestellt werden, um so eine genauere Vorstellung von SIMSCRIPT zu erhalten.

SIMSCRIPT besteht aus den folgenden Grundelementen, welche das simulierte System vollständig erfassen:



Wie in Figur 7-1 wird also das simulierte System durch seinen Zustand und durch Zeitpunkte von Ereignissen, durch die der Zustand geändert wird, beschrieben.

Entities repräsentieren die Objekte in der Realität, z.B. Menschen, Maschinen, Fahrzeuge, Häuser, Informationen etc. Ein simuliertes System kann eine beliebige Zahl verschiedener Typen von Entities aufweisen und die Anzahl Entities vom selben Typ ist nicht restringiert.

Attribute sind die Eigenschaften der Entities und haben Werte. Z.B. können Attribute der Entity Mensch Alter, Geschlecht, Anzahl Kinder, aber auch Haarfarbe etc. sein. Mengen schließlich dienen der Verknüpfung oder in Beziehung setzen verschiedener Entities. So kann eine Menge z.B. alle Entities Mensch mit dem Attribut blond beinhalten, aber auch die Reihe der Patienten, welche beim Arzt auf Behandlung wartet, in letzterem Fall wird auch noch angegeben, in welcher Reihenfolge die Patienten behandelt werden. In SIMSCRIPT kann ein Entity zu einer beliebigen Anzahl Mengen gehören. SIMSCRIPT verlangt also, daß das simulierte System in den Begriffen Entity, Attributen und Mengen beschrieben wird. Der erste Schritt der Programmierung besteht daher in der Spezifizierung aller Entities und ihren Attributen sowie möglichen Mengen, zu denen die Entities gehören können. Wird ein Entity definiert, so bedeutet dies eigentlich die Definition einer Klasse von Dingen, wobei jedes Element dieser Klasse dieselbe Liste von Attributen sowie mögliche Mengenzugehörigkeit aufweist. Einfachheitshalber soll die Klasse als Entity-Typ, die Elemente der Klasse als individuelle Entities bezeichnet werden. Als Beispiel sei angeführt:

Entity-Typ:	Mensch
Attribute:	Gewicht, Alter, Größe
Mengen:	SPÖ, ÖVP, FPÖ, KPÖ

In einigen Fällen ist es möglich die Elemente eines Entity-Typs genau im voraus festzulegen, z.B. die Maschinen einer bestimmten Type in einer Fabrik. Wenn also die Elemente eines Entity-Typs im System verbleiben und daher die Anzahl Elemente a priori bekannt ist, so heißen diese Entities permanent. Handelt es sich hingegen um Elemente, welche ins simulierte System eintreten, bearbeitet werden und wieder austreten, wie z.B. Patienten, welche von einem Rettungswagen befördert werden, so nennt man diese Entities temporär und es besteht keine Restriktion über die Elementeanzahl temporärer Entity-Typen.

Das SIMSCRIPT-Programm wird unterteilt in Subprogramme, welche Ereignissen entsprechen. Diese Ereignisse können wieder andere Subprogramme aufrufen. Ereignisse sind, um es zu wiederholen, Zeitpunkte, zu denen der Zustand des Systems ändert. Ereignisse werden behandelt, als passierten sie plötzlich und ohne Zeiterfordernis. Generell kann man sagen, daß jede Aktivität durch zwei Ereignisse beschrieben werden kann, dem Anfang und Ende des Ereignisses. Ereignisse können nur eintreten, falls bestimmte Bedingung erfüllt sind, z.B. kann ein Rettungswagen nur im Einsatz sein, falls ein Notfall vorher telefonisch angezeigt wurde.

SIMSCRIPT beinhaltet eine Zeitroutine (Ereignisliste), um die Reihenfolge und Zeitpunkte der Ereignisse zu bestimmen. Diese Routine ruft jeweils das entsprechende Subprogramm auf. Die Ereignisroutine wird dann exekutiert und anschließend wieder die Zeitroutine aufgerufen. SIMSCRIPT besitzt Befehle, um Ereignisse zu verursachen oder wieder abzusagen. Jede Ereignisroutine enthält Instruktionen, welche Aktivitäten repräsentieren. Sie kann auch Befehle enthalten, welche festlegen, welche anderen Ereignisse in Zukunft passieren und wann.

SIMSCRIPT erlaubt dem Programmierer die Spezifikation derjenigen Operationen, welche üblicherweise in Ereignis-Routinen durchgeführt werden. Dies beinhaltet Zustandsänderungen, Verursachen oder Absagen zukünftiger Ereignisse, Sammeln von Informationen über den Ablauf, speichern dieser Information, Fällen von Entscheidungen u.a.

Nachdem der Zustand des Systems in Entities, Attributen und Mengen beschrieben wird, kann Zustandsänderung nur bedeuten: Erzeugen oder Vernichten von Entity-Elementen, Änderung des Wertes eines Attributes oder Gewinn oder Verlust der Zugehörigkeit eines Entity-Elements zu einer bestimmten Menge. Aktionen dieser Art sind beschrieben durch SIMSCRIPT-Befehle wie CREATE, DESTROY, READ, LET, FILE und REMOVE. Mit Hilfe der Befehle CAUSE und CANCEL können zukünftige Ereignisse festgelegt werden. Zusätzliche Möglichkeiten sind logische Abfragen in Form von IF-Befehlen, zum Fällen von Entscheidungen sowie Suchbefehle der Art FIND MIN, FIND MAX oder FIND FIRST.

Dem Berechnen statistischer Kenngrößen dienen die Befehle ACCUMULATE und COMPUTE. Der Output kann erfolgen durch Befehle, welche mit analogen FORTRAN-Befehlen übereinstimmen oder aber durch spezielle Output-Routinen.

Zum Abschluß dieses Kapitels soll noch ein rudimentäres SIMSCRIPT-Programm zur Simulation des Notfallsystems angeführt werden. Die angegebenen Befehle sind korrekt, aber es fehlen statistische und Output Routinen sowie weitere nötige Spezifikationen. Im Folgenden bedeuten Großbuchstaben SIMSCRIPT-konforme Befehle, Kleinbuchstaben dienen der Kennzeichnung von Namen, welche auf maximal 4-stellige Worte verkodiert werden müßten, um SIMSCRIPT-kompatibel zu sein.

TIME gibt stets die aktuelle Zeit im Programm an.

EXOGENOUS EVENT START

CREATE Notfallanruf

CREATE Krankenruf

CAUSE Notfallanruf AT TIME

CAUSE Krankenruf AT TIME

DO, FOR J = (1) (8)

CREATE Ambulanz

FILE Ambulanz IN Spital

REPEAT

RETURN

END

ENDOGENOUS EVENT Notfallanruf

CREATE Notfall

LET Tim(Notfall)=TIME

LET Ort(Notfall)=Notfallort

FILE Notfall IN Notfallwarteschlange

CREATE Ambulanzabfahrt

CAUSE Ambulanzabfahrt AT TIME

CAUSE Notfallanruf AT TIME - ALOG (RANDM)/14.

RETURN

END

ENDOGENOUS EVENT Krankenruf

CREATE Kranken

LET Tim(Kranken) = TIME

LET Ort(Kranken) = Abholort

FILE Kranken IN Krankenwarteschlange

CREATE Ambulanzabfahrt

CAUSE Ambulanzabfahrt AT TIME

CAUSE Krankenruf AT TIME -ALOG (RANDM)/100.

RETURN

END

ENDOGENOUS EVENT Ambulanzabfahrt
DESTROY Ambulanzabfahrt
IF Spital IS EMPTY, RETURN
IF Notfallwarteschlange IS EMPTY, GOTO 100
REMOVE FIRST Notfall FROM Notfallwarteschlange
REMOVE FIRST Ambulanz FROM Spital
DESTROY Notfall
FILE Ambulanz IN Einsatz
CREATE Ambulanzankunft
CAUSE Ambulanzankunft AT TIME + Einsatzdauer
RETURN

100 -IF Spital IS EMPTY, RETURN
REMOVE FIRST Kranken FROM Krankenwarteschlange
REMOVE FIRST Ambulanz FROM Spital
DESTROY Kranken
FILE Ambulanz IN Einsatz
CREATE Ambulanzankunft
CAUSE Ambulanzankunft AT TIME + Einsatzdauer
RETURN
END

ENDOGENOUS EVENT Ambulanzankunft
DESTROY Ambulanzankunft
CREATE Ambulanzabfahrt
CAUSE Ambulanzabfahrt AT TIME
REMOVE FIRST Ambulanz FROM Einsatz
FILE Ambulanz IN Spital
RETURN
END

EXOGENOUS EVENT END
STOP
END

7.4. Vergleich verschiedener Simulationssprachen

Es sollen nun einige gängige Simulationssprachen kurz erläutert werden, wobei insbesondere die folgenden Kriterien zur Beurteilung einer Simulationssprache wesentlich erscheinen:

- Erleichterung bei Modell-Formulierung
- Einfachheit beim Programmieren
- Gut ausgebaute Fehlerdiagnose
- Breite des Anwendungsspektrums

FORTRAN, ALGOL und PL/I sollen der Vollständigkeit halber aufgeführt werden, da grundsätzlich jedes Simulationsmodell in diesen Sprachen geschrieben werden kann. Allerdings ist nur die Breite des Anwendungsspektrums positiv zu beurteilen.

GASP ist eine Menge von Subroutinen in FORTRAN, welche mit FORTRAN kombiniert werden können und gewisse Unterstützung beim Programmieren bieten.

SIMSCRIPT und SIMULA sind die Sprachen mit dem breitesten Anwendungsbereich bei diskreten Simulationsmodellen.

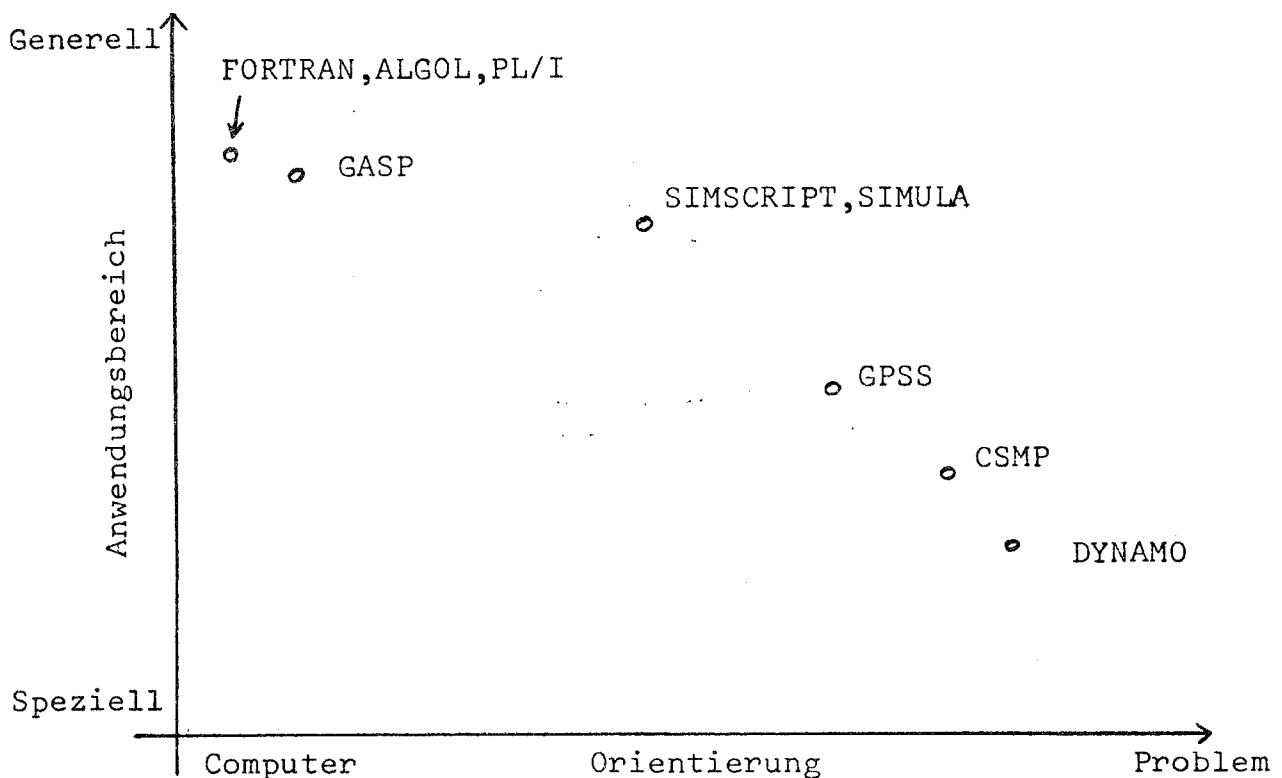
SIMSCRIPT basiert auf FORTRAN und SIMULA auf ALGOL. Allerdings ist SIMULA nur sehr beschränkt verbreitet und hat den Nachteil, daß der Aufwand zum Erlernen dieser Sprache um einiges größer als für SIMSCRIPT ist.

GPSS ist die wohl verbreitetste Sprache für diskrete Simulationsmodelle, allerdings spezieller als SIMSCRIPT und SIMULA, da primär auf komplexe Warteschlangenmodelle zugeschnitten. Dementsprechend ist sie für Warteschlangenmodelle sehr benützerfreundlich, sowohl was die Erleichterung der Modellformulierung als auch Einfachheit des Programmierens betrifft.

DYNAMO und CSMP sind Sprachen für jene kontinuierliche Simulationsmodelle, welche in Form von speziellen Differential-

oder Differenzengleichung geschrieben werden können. Beide erlauben, eine Art Analogrechner auf einem Digitalrechner nachzubilden. CSMP wird vor allem bei technischen Problemstellungen, DYNAMO für betriebliche oder ökonomische Simulationsmodelle verwendet. Das Anwendungsspektrum der Sprachen ist nicht groß, aber für die speziellen Fragestellungen sind sie sehr benutzerfreundlich. Zu DYNAMO wäre aber noch zu bemerken, daß die Methode zum Lösen der Differenzengleichungen im Vergleich zu anderen Integrationsmethoden (z.B. Simpsonregel) sehr schlicht und dementsprechend bei größeren Problemen numerisch ungünstig ist.

Vergleicht man die erwähnten Sprachen nochmals in Bezug auf Problemorientierung und Anwendungsbereich so ergibt sich etwa Figur 7-2:



Figur 7-2

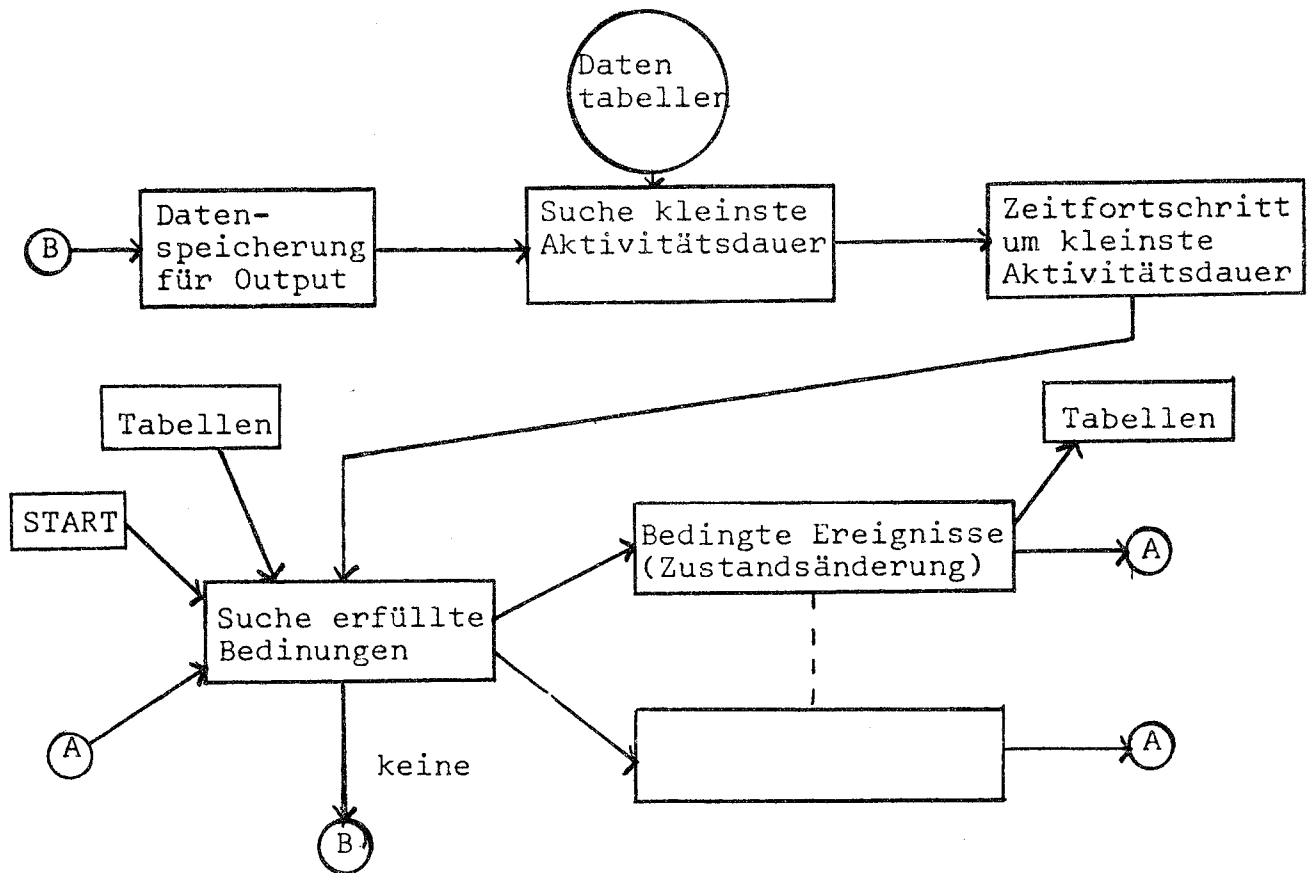
Etwas ausführlicher sei noch das Prinzip besprochen, das den beiden Programmiersprachen CPS und CSL zugrunde liegt, die für diskrete Simulationsmodelle, aber anderen Typs, als bisher besprochen, konzipiert sind.

Figur 7-1 zeigt die Struktur eines Programms, welches ereignisorientiert ist. Das heißt, daß das System betrachtet wird als eine Abfolge von Ereignissen, welche den Systemzustand verändern. Daher initiiert die Ereignisliste das dynamische Verhalten des Systems.

Ein wichtiger Teil des Prozesses in Figur 7-1 ist die Zustandsänderungskontrolle. Dabei wird der Zustand daraufhin untersucht, ob ein bedingtes Ereignis stattfinden kann oder nicht. Nun ist eine wichtige Annahme in ereignisorientierten Simulationen, daß es nur wenige bedingte Ereignisse gibt. Daher ist diese Vorgangsweise dann von Vorteil, falls im System viele Entity-Elemente relativ einfachen Prozessen unterworfen werden (z.B. viele Patienten werden im Notfallsystem abgeholt und abgeliefert). Diese Art von diskreten Simulationsmodellen wird auch material- oder elementorientiert genannt. Der typische Befehl in SIMSCRIPT zum Bestimmen eines bedingten Ereignisses ist

IF Menge IS (NOT) EMPTY, Befehl.

In manchen Systemen ist nun die Bedingung der elementorientierten Simulation (viele Entity-Elemente, wenige Prozesse) nicht erfüllt -so bei der Herstellung eines komplizierten Produkts, bei der ein einziges Stück sehr komplexe, zusammenhängende Arbeitsgänge durchläuft. In diesen Fällen müssen viele Bedingungen erfüllt sein, damit eine Aktivität starten kann. Es ist daher naheliegend, für solche Fälle eine Simulationssprache zu verwenden, welche aktivitätsorientiert ist. Die Steuerungszentrale des Programms ist dann nicht die Ereignisliste, sondern die Bedingungsliste. Die entsprechende Programmlogik ist in Figur 7-3 dargestellt.



Figur 7-3

Aktivitätsorientierter Prozeß

Gemäß Figur 7-3 werden Zustandsänderungen über Tabellen bewirkt, welche Daten über jedes Entity-Element enthalten. Beim Start einer Aktivität wird stets für jedes Entity-Element die relative Prozeßdauer, welche zum Vervollständigen seines Anteils an der Aktivität nötig ist, gespeichert. Der Zeitfortschritt erfolgt dann, indem alle Entity-Elemente abgesucht werden auf kürzeste Aktivitätsdauer. Der Vorteil einer solchen Programmstruktur ist:

- es ist einfacher, das nächste bedingte Ereignis festzuhalten, falls mehrere gleichzeitig aktiviert werden
- eine Hierarchie bedingter Aktivitäten kann einfacher behandelt werden

- es können jene Fälle besser behandelt werden, wo eine Aktivität die Dauer einer anderen, früher gestarteten beeinflussen kann.

Die wichtigste aktivitätsorientierte Programmiersprache ist CSL.

7.5. Literatur

SIMSCRIPT-Manual (Jede Computerfirma bietet ihr eigenes SIMSCRIPT-Manual an, weswegen kein bestimmtes referiert wird. Dies gilt auch für die anderen erwähnten Simulations-sprachen).

Kivait, P.J., R.Villanueva & H.M.Markowitz, The Simscript II Programming Language, Prentice Hall, 1969

Teichroew, D. & J.F.Lubin, Computer Simulation - Discussion of the Technique and Comparison of Languages, Communications of the ACM, Vol.9 (1966), pp.723-741.

Tocher, K.D., Review of Simulation Languages, OR Quarterly, Vol.16 (1965), pp.189-218.

8. ANALYSE VON SIMULATIONSRUNS

8.1. Einleitung

In diesem Kapitel sollen die Methoden zur Analyse vor allem stochastischer und instationärer Simulationsmodelle genauer erörtert werden. Der größte Teil des folgenden Stoffes gehört in das Gebiet der stochastischen Prozesse und wird hier nur rudimentär behandelt. Nach einführenden Definitionen wird das Problem des Einpendelns von Simulationsruns aus einem transienten Anfangszustand in einen stationären Zustand erörtert. Daran schließt sich ein Exkurs über die Behandlung instationärer Vorgänge an.

8.2. Definitionen

Unter einem Simulationsrun versteht man einen ununterbrochenen Programmablauf und Datenspeicherung eines Simulationsprogramms bei einer speziellen aber unveränderten Kombination kontrollierbarer Variabler. Eine Wiederholung eines Runs ist ein Run mit gleicher Variablenkombination, aber einer anderen Zufallszahlenfolge. Eine Beobachtung eines simulierten Systems ist ein Abschnitt eines Runs, welcher zum Schätzen der gewünschten Kenngrößen des Systems dient. (Die Beobachtung passiert also nicht in einem Zeitpunkt, sondern hat eine zeitliche Ausdehnung.)

Wir nennen den Zustand eines Systems stabil oder stationär, falls aufeinanderfolgende Beobachtungen des Systemablaufes statistisch ununterscheidbar sind. In anderen Worten bedeutet stationär einen Zustand in dem aus später folgenden Beobachtungen keine neue Information über das Verhalten des Systems gewonnen werden können.

Es ist zu beachten, daß die Definition der Stationarität sich von der streng mathematischen unterscheidet. Bei dieser wird nämlich verlangt, daß der Verteilungsfunktionsvektor $g(t)$,

welcher das Verhalten der Systemkenngrößen im Zeitpunkt t beschreibt, identisch mit $g(t+\delta)$ für alle $\delta > 0$ ist. Diese Definition ist jedoch zu restriktiv, da durch sie Systeme ausgeschlossen werden, welche ein genau vorhersagbares zyklisches Verhalten aufweisen, wohingegen unsere Definition Zyklen zuläßt, indem die Beobachtungsdauer genau eine Zykluslänge umfaßt.

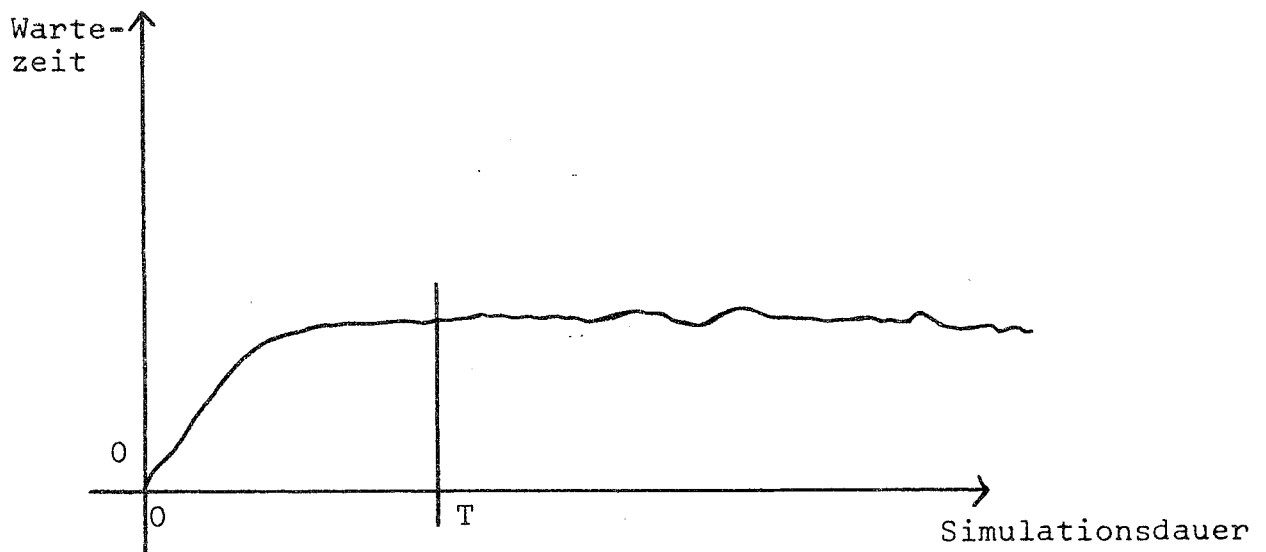
Ein System welches kein stationäres Verhalten zeigt, nennt man instationär oder den entsprechenden Zustand transient.

Transiente Zustände ergeben sich auch bei grundsätzlich stationären Systemen zum Anfang der Simulation, da üblicherweise der extern definierte Anfangszustand nicht mit dem systemeigenen stationären Zustand überein stimmt, weswegen das System bis zum Erreichen des stationären Zustands instationäres Verhalten aufweist. Oft interessieren aber gerade die transienten Phänomene oder aber das System ist grundsätzlich instationär, wie es z.B. bei einem Simulationsmodell zu erwarten ist, welches makroökonomische Vorgänge prognostizieren soll.

8.3. Methoden zur Ausschaltung unerwünschter transienter Zustände

Simulationsprogramme, deren Aufgabe es ist, stationäres Verhalten zu studieren, weisen, wie bereits erwähnt, auf Grund der Startbedingungen transiente Zustände auf. Diese transienten Zustände sind nun für die Analyse unerwünscht, weswegen die Frage entsteht, wie sie vermieden oder wenigstens minimal (im Sinne einer kurzen Zeitspanne) gehalten werden können. Eine erste Möglichkeit besteht darin, den Anfangszustand möglichst dem stationären Zustand anzunähern, was allerdings nur beschränkt möglich ist, da der stationäre Zustand nicht a priori bekannt ist. Wäre er es, ersparte man sich das Simulationsmodell. Allerdings entsteht bei diesem Vorgehen das Problem, was zu tun ist, falls zwei verschiedene Runs

(also mit verschiedenen Werten der Kontrollvariablen) durchgeführt und verglichen werden sollten. Paßt man den Anfangszustand dem jeweiligen stationären Zustand an, wird ein Vergleich zumindestens problematisch. Pragmatisches Vorgehen in diesem Fall wäre ein wohl für beide Runs fester Anfangszustand, welcher aber in etwa ein Mittel zwischen beiden stationären Zuständen darstellt. Aber auch bei günstiger Wahl des Anfangszustandes wird zu Beginn stets transientes Verhalten zu beobachten sein. Die Effekte transientser Zustände dürfen die Beobachtung des stationären Zustand nicht beeinflussen. Nehmen wir als Beispiel beim Ambulanzmodell die Wartezeit zwischen Notfallanruf und Abfahrt eines Ambulanzautos. Dann wird diese Wartezeit, abhängig von der Simulationsdauer, bei einem Anfangszustand des Systems ohne Warteschlange bei Notfällen und Kranken und kein Ambulanzfahrzeug im Einsatz, gemäß Figur 8-1 sich darstellen.



Figur 8-1

Bricht man nun im Zeitpunkt T den Simulationslauf ab und schätzt die erwartete Wartezeit aus der Beobachtung im Zeitintervall $[0, T]$, so erkennt man leicht wie stark das Resultat verfälscht sein wird.

Um diesen unerwünschten Effekt einzudämmen, ergeben sich zwei Möglichkeiten. Man kann eine sehr lange Simulationsdauer wählen (mit dem damit verbundenen Rechenaufwand) oder die Beobachtung erst einige Zeit nach Simulationsstart beginnen. Erstere Methode wird allerdings den Effekt transienter Zustände nie beseitigen sondern für immer längere Simulationsdauer immer mehr verkleinern, wohingegen letztere Methode bei richtiger Wahl von T (siehe Figur 8-1) transiente Effekte ausschließt. Allerdings entsteht nun wiederum das Problem, wie man feststellt ob ein Zustand stationär geworden ist, oder ob dies noch nicht der Fall ist.

Es existieren keine verlässlichen Methoden die Stationarität festzustellen, jedoch zwei gebräuchliche Verfahren, die die Stationarität nahelegen.

Bei der ersten Methode berechnet man jeweils aus den N letzten Zeitpunkten das arithmetische Mittel der realisierten Kenngrößenwerte. Sei dieser Vektor $E(t)$ zum Zeitpunkt t. Gilt nun, daß über mehrere Zeitpunkte hinweg $E(t)$ gleich bleibt, daß also

$$E(t) = E(t+d) \quad d = 1, 2, \dots, r \quad (8-1)$$

gilt, so kann man daraus heuristisch folgern, daß das System seit dem Zeitpunkt $t-N$ stationär ist.

Die zweite Methode beruht ebenfalls darauf, daß man den Vektor $E(t)$ bestimmt, aber nun testet, ob für die folgenden Zeitpunkte die Häufigkeit mit der ein realisierter Kenngrößenwert über $E(t)$ liegt dieselbe ist, mit der er unter $E(t)$ liegt. Ist dies der Fall, so nimmt man ebenfalls Stationarität an.

8.4. Die Verwendung von Simulationsmodellen zur Analyse transienter Phänomene

Wie bereits in Kapitel 2.4. angedeutet, interessieren auch Simulationsmodelle, welche gerade instationäre Prozesse studieren helfen sollen. Entweder ist der betrachtete Prozeß grundsätzlich instationär oder man interessiert sich gerade für die transienten Zustände, welche beim Übergang vom Anfangszustand in den stationären Zustand auftreten. Nicht selten sind stationäre Zustände auch mathematisch-analytisch angebar und die transienten Zustände nicht. Bedeutung gewinnen transiente Zustände beim praktischen Einführen eines neuen Prozesses (z.B. neue Lagerhaltungsstrategie), oder aber wenn der Prozeß von so kurzer Dauer ist, daß er den stationären Zustand gar nicht erreicht (z.B. Aufbau einer Warteschlange beim Arzt mit kurzer Ordinationszeit).

Der Simulationsoutput zur Beschreibung transienter Phänomene ist nicht unbedingt dergleiche, wie der zur Beschreibung stationärer Zustände. Bei stationären Prozessen interessiert meist der Erwartungswert oder typische Ereignisse beim Prozeßablauf. Bei instationären Systemen ist es oftmals wichtiger, die Extremsituationen zu erfassen, also Maximal- und Minimalwerte und deren Wahrscheinlichkeit bei den Kenngrößen.

Vom Programmablauf her ergeben sich ebenfalls Unterschiede. Analysen stationärer Prozesse können genauer werden, indem die Simulationsdauer vergrößert wird. Instationäre Prozesse müssen stets vom Startpunkt an wiederholt werden, um bessere Aussagen zu erhalten. Weiters ist es i.a. nötig, zum Studium der Extremsituation mehr Beobachtungen durchzuführen, als für die Analyse typischer Ereignisse.

8.5. Methode zur Analyse stochastischer Simulationsruns

Bereits in Kapitel 3.6. wurde dargelegt, wie man den Erwartungswert von unabhängigen Stichproben und die Varianz der Schätzung bei stationären Prozessen schätzen kann.

Auf Grund der Kenntnis, daß der geschätzte Erwartungswert \bar{X} für große N normalverteilt ist, läßt sich auch ein Vertrauensintervall für X angeben. Mit den Bezeichnungen aus Kapitel 3.6. gilt nämlich

$$P \left\{ \frac{\bar{X}^{(N)} - \mu}{\frac{S}{\sqrt{N}}} \leq u \right\} \approx \phi(u) \quad (8-2)$$

wo $\phi(u)$ die Verteilungsfunktion der Normalverteilung darstellt. Es gilt daher auch

$$P \left\{ -u \leq \frac{\bar{X}^{(N)} - \mu}{\frac{S}{\sqrt{N}}} \leq u \right\} \approx \phi(u) - \phi(-u) \quad (8-3)$$

was man, wegen $\frac{S}{\sqrt{N}} > 0$, umformen kann zu

$$P \left\{ \bar{X}^{(N)} - \frac{S}{\sqrt{N}} u \leq \mu \leq \bar{X}^{(N)} + \frac{S}{\sqrt{N}} u \right\} \quad (8-4)$$

Wählt man nun u als Lösung der Gleichung

$$\phi(u) - \phi(-u) = W$$

die, wegen der Symmetrie der Normalverteilung auch geschrieben werden kann als

$$\phi(u) = \frac{1+W}{2}$$

mit vorgegebenem W , so stellt (8-4) ein W -Vertrauensintervall für μ dar. Die Breite dieses Intervalls beträgt genähert

$$2 \cdot \frac{S}{\sqrt{N}} u$$

Man benötigt daher, um die Schätzgenauigkeit von $\bar{X}^{(N)}$ um eine Dezimalstelle zu erhöhen $100 \times (N-1)$ weitere Stichproben.

Da dieser Rechenaufwand doch groß ist, wollen wir nun die Frage untersuchen, ob man durch geschickte Wahl der Zufallszahlenfolge die Varianz reduzieren kann. Hierbei soll aber weiterhin Unabhängigkeit der Stichprobenwerte vorausgesetzt werden.

Untersuchen wir den Fall, daß ein Run wiederholt wird, aber eine Korrelation zwischen den zwei Runs besteht (die Beobachtungen innerhalb eines Runs sind unabhängig). Seien die beobachteten Werte im ersten Run X_i^1 und im zweiten Run X_i^2 , beide mit Varianz σ^2 , wobei die Anzahl Beobachtungen in beiden Runs dieselbe sein soll, nämlich $n/2$, und sei der Korrelationskoeffizient zwischen den Beobachtungen im ersten und zweiten Run g .

Dann gilt

$$\bar{X} = \frac{1}{n/2} \sum_{i=1}^{n/2} \frac{(X_i^1 + X_i^2)}{2}$$

und für die Varianz von \bar{X}

$$S^2 = \frac{\sigma^2}{n} (1+g) \quad (8-5)$$

Für unabhängige Runs ist $g=0$ und das Resultat ident mit dem bei einem einzigen Run mit n Beobachtungen analog zu Kapitel 3.6. Für negative Korrelationen $g < 0$, (d.h. falls X_i^1 klein, ist X_i^2 groß und umgekehrt), läßt sich gemäß (8-5) die Varianz reduzieren.

Eine übliche Methode negativ korrelierte Zufallszahlen zu erzeugen besteht darin, bei in $[0,1]$ gleichverteilten Zufallszahlen R_i^1 im ersten Run, diese durch eine Zufallszahlenfolge

$$R_i^2 = 1 - R_i^1$$

im zweiten Run zu ersetzen.

Es soll nun untersucht werden, wie zwei Runs mit verschiedenen Werten der kontrollierbaren Variablen verglichen werden können. Angenommen der Erwartungswert wurde in beiden Runs durch \bar{X}_1 und \bar{X}_2 geschätzt mit den zugehörigen Varianzen der Schätzung S_1^2 und S_2^2 . Zum Vergleich der zwei Runs interessiere nun die Differenz der Erwartungswerte, dann gilt

$$\begin{aligned}\bar{X}_D &= \bar{X}_1 - \bar{X}_2 \\ S_D^2 &= S_1^2 + S_2^2\end{aligned}\tag{8-6}$$

Führt man nun wiederum eine Korrelation g zwischen den beiden Runs ein, so gilt

$$S_D^2 = S_1^2 + S_2^2 - 2gS_1 \cdot S_2\tag{8-7}$$

weswegen für positiv korrelierte ($g > 0$) Runs die Varianz abnimmt. Das bedeutet, daß wenn man beide Runs mit den gleichen Zufallszahlen für gleiche Ereignisse durchführt, ohne Erhöhung der Simulationsdauer die Varianz reduziert wird.

Aus dem Gesagten könnte nun die Vermutung entstehen, daß es stets besser sei mehrere kurze, aber korrelierte Runs durchzuführen, statt einen langen. Daß dem nicht so ist, ergibt sich daraus, daß zu Beginn eines Runs stets eine transiente Phase existiert, die bei Wiederholung von Runs ebenfalls wiederholt wird und, falls diese Phase lang ist, der Gewinn durch korrelierte Runs aufgehoben wird durch mehrfache transiente Phasen, die keinerlei Informationen liefern. Varianzreduktion durch korrelierte Runs wird daher wohl nur bei Simulationen mit kurzer transienter Phase Einsparung an Computerzeit erbringen.

Bislang haben wir stets angenommen, daß die Beobachtungen innerhalb eines Runs unabhängig sind. Nur unter diesen Voraussetzungen konnten Erwartungswert und Varianz geschätzt werden. Wir wollen

diese Annahme nun fallen lassen und untersuchen, wie man einen stationären Prozeß, dessen Beobachtungen nicht unabhängig voneinander sind (sogenannter autokorrelierter Prozeß), analysieren kann.

Läßt sich über den autokorrelierten Prozeß weiter nichts annehmen, so muß, um etwas über die Varianz des geschätzten Erwartungswertes aussagen zu können, die Autokorrelationsfunktion geschätzt werden. Dies geschieht indem man schätzt

$$\text{Cov}(X_t, X_{t+s}) = C_s \approx \frac{E\{ [X_t - \bar{X}][X_{t+s} - \bar{X}] \}}{\sigma^2} \quad (8-8)$$

wobei

X_t

Beobachtung zum Zeitpunkt t

$$\bar{X} = \frac{\sum_{t=1}^N X_t}{N}$$

Geschätzter Erwartungswert über X_t , $t=1,2,\dots$

σ^2

Varianz von X_t , $t=1,2,\dots$

Die Autokorrelationsfunktion C_s in (8-8) ist keine Funktion von t , sondern nur von der Differenz s zweier Zeitpunkte, da der Prozeß stationär ist. C_s muß nun für alle Werte $s > 0$ geschätzt werden. Dann gilt

$$\text{Var } \bar{X} = \frac{1}{N^2} \text{Var} \left\{ \sum_{t=1}^N X_t \right\} = \frac{1}{N^2} \left[\sum_{t=1}^N \text{Var} X_t + 2 \sum_{1 \leq t < u \leq N} \text{Cov}(X_t, X_u) \right] \quad (8-9)$$

Aber, da der Ausdruck in eckigen Klammern in (8-9) wegen (8-8) geschrieben werden kann als

$$\begin{aligned} & C_0 + C_1 + C_2 + \dots + C_{N-1} + \\ & + C_1 + C_0 + C_1 + \dots + C_{N-2} + \\ & + C_2 + C_1 + C_0 + \dots + C_{N-3} + \\ & \vdots \\ & \vdots \\ & \vdots \\ & \vdots \\ & + C_{N-1} + C_{N-2} + C_{N-3} + C_0 \end{aligned}$$

ergibt sich

$$\text{Var } \bar{X} = \frac{1}{N} C_0 + 2 \sum_{t=1}^{N-1} \frac{t}{N} C_{N-t} \quad (8-10)$$

Für den unkorrelierten Fall, also $C_t = 0$ für $t \neq 0$, ergibt sich aus (8-10) die Beziehung bei unabhängigen Beobachtungen als Spezialfall.

Da das Schätzen von (8-8) sehr aufwendig ist, kann man im Falle eines regenerativen Prozesses eine andere Methode benutzen. Man sagt, ein stochastischer Prozeß ist regenerativ, falls es eine Folge von zufälligen Zeitpunkten

$$0 < b_1 < b_2 < \dots \quad (8-11)$$

gibt, sodaß zu jedem dieser Zeitpunkte der Prozeß aufs Neue beginnt und seine Entwicklung denselben probabilistischen Gesetzen gehorcht, denen er schon zum Zeitpunkt b_1 gehorchte. Der Prozeß zerfällt mithin in gegenseitig unabhängige, probabilistisch äquivalente Blöcke zufälliger Dauer.

Ist

$$b_0 = 0$$

$$Y_{i,2} = b_i - b_{i-1} \geq 0 \quad i = 1, 2, \dots$$

so soll gelten

$$E Y_{i,2} < \infty$$

d.h. die mittlere Dauer eines Blocks sei endlich. Die Zeitpunkte b_i werden Regenerationszeitpunkte genannt und aus ihrer Existenz folgt die Stationarität des Prozesses. Alles, was zwischen den Zeitpunkten b_{i-1} und b_i passiert, wird mit i -tem Zyklus bezeichnet. Beobachtungen aus zwei verschiedenen Zyklen sind unabhängig.

Dann gilt mit

$$Y_{i1} = \sum_{t=b_{i-1}}^{b_i} X_t, \quad Y_{i2} = b_i - b_{i-1} \quad i=1,2,\dots$$

wo X_t die Beobachtung zum Zeitpunkt t ist, daß

$$EX_t = \frac{E(Y_{i1})}{E(Y_{i2})} \quad (8-12)$$

Hat man nun einen regenerativen Prozeß über N Zyklen beobachtet, so liegen N unabhängige Ergebnispaare

$$(Y_{i1}, Y_{i2}) \quad i=1, \dots, N$$

vor und (8-12) läßt sich schätzen durch

$$E(X_t) = \frac{E(Y_{i1})}{E(Y_{i2})} = \frac{\mu_1}{\mu_2} \approx \frac{\sum_{i=1}^N Y_{i1}}{\sum_{i=1}^N Y_{i2}} \quad (8-13)$$

(8-13) liefert natürlich die übliche Schätzfunktion für $E(X_t)$. Es fehlt nun aber die Angabe über die Genauigkeit der Schätzung. Es soll nun ein Verfahren zum Bestimmen des Vertrauensintervalls des Erwartungswertes von $E(t)$ angegeben werden.

Das Problem lautet also: Seien $(Y_{11}, Y_{12}), (Y_{21}, Y_{22}), \dots, (Y_{N1}, Y_{N2})$ gegenseitig unabhängige zweidimensionale Zufallsvektoren gleicher Verteilungsfunktion

$$\vec{Y}_i = \begin{bmatrix} Y_{i1} \\ Y_{i2} \end{bmatrix}$$

mit unbekanntem Erwartungswertvektor

$$E \vec{X}_i = \vec{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$$

und unbekannter positiv-definiten Kovarianzen-Matrix

$$\underline{\sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{12} & \sigma_{22} \end{bmatrix}$$

Sei

$$v = \frac{\mu_1}{\mu_2}$$

Für v sucht man ein Vertrauensintervall ausgehend von N Beobachtungen $\vec{Y}_1, \dots, \vec{Y}_N$.

Der Stichprobenmittelwert ist:

$$\vec{\bar{Y}}(N) = \frac{1}{N} \sum_{i=1}^N \vec{Y}_i = \begin{bmatrix} \bar{Y}_1(N) \\ \bar{Y}_2(N) \end{bmatrix}$$

und die Stichproben-Streuungs matrix

$$\underline{s}(N) = \frac{1}{N} \sum_{i=1}^N (\vec{Y}_i - \vec{\bar{Y}}(N)) (\vec{Y}_i - \vec{\bar{Y}}(N))^*$$

worin $*$ das Transpositionszeichen ist.

$$\underline{s}(N) = \begin{bmatrix} s_{11}^{(N)} & s_{12}^{(N)} \\ s_{21}^{(N)} & s_{22}^{(N)} \end{bmatrix}$$

Betrachtet man die neuen Zufallsgrößen

$$Z_i = Y_{i1} - v Y_{i2} \quad i=1, \dots, N$$

so gilt für sie

$$E Z_i = 0 ; \text{Var } Z_i = (\sigma_Z)^2 = \sigma_{11} - 2v\sigma_{12} + v^2\sigma_{22}$$

Sei nun

$$\bar{Z}(N) = \frac{1}{N} \sum_{i=1}^N Z_i$$

Gemäß zentralem Grenzwertsatz gilt für $N \gg 1$ genähert

$$P \left\{ \frac{\sqrt{N}}{\sigma_Z} \bar{Z}(N) \leq u \right\} = \Phi(u) \quad (8-14)$$

worin $\Phi(u)$ die Verteilungsfunktion der standardisierten Gauss-Zufallsgröße ist.

Die Größe

$$(s_Z(N))^2 = s_{11}(N) - 2vs_{12}(N) + v^2s_{22}(N)$$

ist ein Schätzwert von σ_Z^2 , der gemäß des Gesetzes der großen Zahlen mit Wahrscheinlichkeit Eins gegen den richtigen Wert konvergiert.

Es ist deshalb statthaft, für $N \gg 1$ (8-14) genähert zu schreiben

$$P \left\{ \frac{\sqrt{N}}{s_Z(N)} \bar{Z}(N) \leq u \right\} \approx \Phi(u) .$$

Es gilt daher ebenfalls

$$P \left\{ -u \leq \frac{\sqrt{N}}{s_Z(N)} \bar{Z}(N) \leq u \right\} \approx W = \Phi(u) - \Phi(-u) \quad (8-15)$$

mit u als Lösung der Gleichung

$$\Phi(u) = \frac{1+W}{2}$$

mit vorgegebenem W (z.B. 0.9 oder 0.99,...).

Setzt man zur Abkürzung

$$k = \frac{u^2}{N}, \quad (8-16)$$

so läßt sich (8-15) schreiben, als

$$P\{(\bar{Z}(N))^2 - k \cdot (s_Z(N))^2 \leq 0\} \approx W.$$

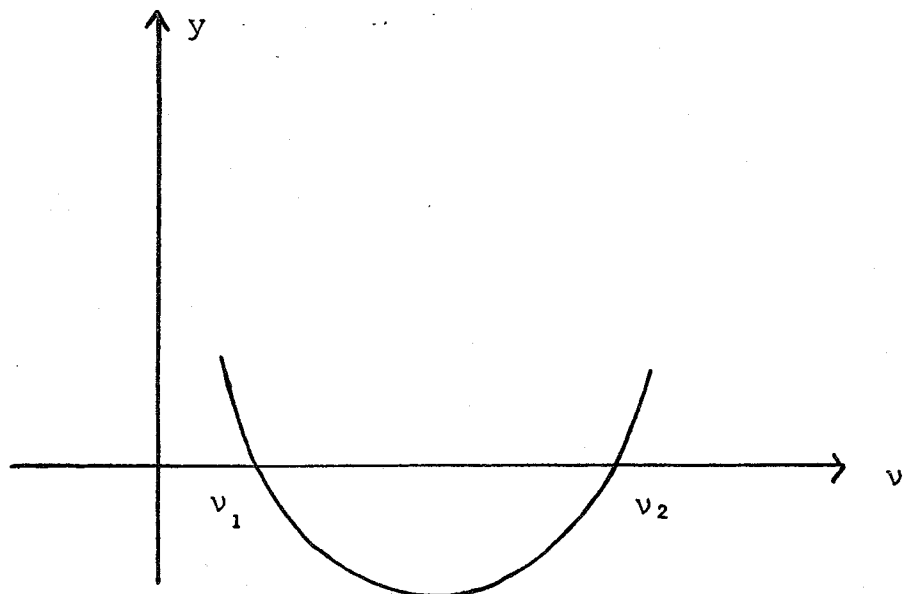
Der Ausdruck in geschweiften Klammern, der nicht-positiv sein soll, ist ein quadratisches Polynom in v , und zwar

$$y = [(\bar{Y}_2^2 - k \cdot s_{22})] v^2 - 2[\bar{Y}_1 \bar{Y}_2 - k s_{12}] v + [(\bar{Y}_1)^2 - k s_{11}] \quad (8-17)$$

Es hat also die Form

$$y = a \cdot v^2 + 2b v + c$$

Aus (8-16) folgt, daß a für genügend große Werte von N fast immer positiv ist, deshalb stellt (8-17) immer eine nach oben geöffnete Parabel dar, wie Figur 8-2 zeigt.



Figur 8-2

Das interessierende Ereignis $\{y \leq 0\}$ ist also äquivalent mit $\{v_1 \leq v \leq v_2\}$, wie man sich aus der Zeichnung leicht überzeugt. Hiemit liegt das gesuchte W-Vertrauensintervall für v vor.

Zu den beiden besprochenen Methoden, autokorrelierte stationäre Prozesse zu analysieren, ist abschließend zu sagen, daß bei der praktischen Durchführung der Methoden doch einige Probleme entstehen. Im ersten Fall ist das Schätzen von (8-8) problematisch, da ungeheuer datenintensiv, im anderen Fall ist das Bestimmen der Regenerationszeitpunkte problematisch. Bei genauer Kenntnis der Autokorrelationsfunktion einerseits und der Regenerationszeitpunkte andererseits wird bei gleich langer Beobachtungsdauer für beide Methoden die erstere mit (8-9) eine kleinere Varianz liefern als die zweite mit (8-17), da die Autokorrelationsfunktion eine genauere Beschreibung des Prozesses liefert und aus ihr auch die Regenerationszeitpunkte bestimmt werden können.

8.6. Anwendung auf Ambulanzsystem

Wir wollen nun daran gehen, das Computerprogramm des Ambulanzmodells zu erstellen. Zum Vergleich wird am Schluß von Kapitel 8 je ein Programm in FORTRAN und eines in SIMSCRIPT aufgelistet. Es soll nun genau spezifiziert werden, wie die Struktur des Programms, insbesondere die statistischen Analysen und Abbruchkriterien aussehen soll. Nun, die Struktur des Programms wurde bereits als rudimentäres SIMSCRIPT-Programm angegeben, die Input-Daten sind ebenfalls spezifiziert. Als wichtigste Outputgröße haben wir bereits die Antwortzeit=Wartezeit+Fahrzeit festgehalten. Wir können noch weiter einengen, daß die Fahrzeit in unserem System, wo ja alle Fahrzeuge von derselben Einsatzstelle abfahren, durch den Entscheidungsparameter, nämlich die Anzahl Einsatzfahrzeuge, nicht beeinflusst wird. Anders wäre die Situation, wenn verschiedene Standorte von Einsatzstellen zur Diskussion stünden. Somit ist die wirklich relevante Größe im System die Wartezeit des Notfalls, das ist die Zeit, die zwischen Notfalleinruf und Ambulanzabfahrt von der Einsatzstelle vergeht. Diese Wartezeit wird Null sein, falls so viele Ambulanzen zur Verfügung stehen, daß stets eine Ambulanz in der Einsatzstelle steht, die Wartezeit wird hingegen beliebig

wachsen, falls weniger Ambulanzen vorhanden sind, als Notfallanrufe bedient werden können. In diesem Falle wird das System also instationär. Nun dürfen aber die Kranken nicht außer acht gelassen werden, denn obwohl Notfall stets Priorität vor Kranken hat, und die Wartezeit des Notfalls das wichtigste Kriterium ist, muß doch die Wartezeit des Kranken innerhalb gewisser Grenzen bleiben, insbesondere darf nicht der ebenfalls instationäre Fall eintreten, daß zwar die Wartezeit des Notfalls einen stationären Wert annimmt, hingegen die Wartezeit des Kranken wieder beliebig wächst. Dieser Fall könnte eintreten, falls gerade so viele Ambulanzen zur Verfügung stehen, um eben noch die Notfälle zu bedienen. Wir wollen daher als Output des Systems geschätzten Erwartungswert und Verteilungsfunktion der Wartezeiten von Notfall und Kranken sowie, um etwas über die Güte der Schätzung auszusagen, die Schätzung der Standardabweichung des geschätzten Erwartungswertes. Weiters soll das System selbst feststellen, ob instationäres Verhalten vorliegt, in Bezug auf die Wartezeiten - ein anderes instationäres Verhalten kann es bei diesem System nicht geben. Als zusätzliche Information wäre noch sinnvoll: die mittlere Auslastung jedes Fahrzeuges sowie zum Überprüfen der Korrektheit des Programms, die mittlere Fahrzeit mit und ohne Blaulicht und die mittlere Anruhfrequenz von Kranken und Notfällen. Zum Ausschalten der transienten Zustände zu. Beginn des Modells soll die Methode der gleitenden Mittelwertberechnung dienen, mit deren Hilfe auch instationäres Verhalten erkannt werden kann. Das Programm soll schließlich die entsprechenden Wartezeiten bei vorgegebener Anzahl Ambulanzen bestimmen, wobei auch festzustellen ist, ob und bei welcher Anzahl instationäres Verhalten auftritt und wieviele Ambulanzen nötig sind, um die Wartezeiten auf Null zu reduzieren.

Sehr einfach kann man sich auf Grund der vorgegebenen Daten eine grobe Abschätzung ermitteln für die minimale Anzahl Ambulanzen, welche für stationäres Verhalten nötig sind.

Im Mittel rufen 114 Notfälle und Kranke zusammen pro Tag, also in 24 Stunden an. Die mittlere Fahrzeit beträgt grob 15 Minuten, sodaß eine Ambulanz ca. 30 Minuten benötigt um einen Notfall oder Kranken zu bedienen. Also kann eine Ambulanz 2 Anrufe pro Stunde bedienen. Es kommen aber $114/24 > 4$ Anrufe pro Stunde, weshalb 2-3 Ambulanzen mindestens nötig sein werden.

Wie wir gezeigt haben, bereitet die Schätzung der Standardabweichung geschätzter Erwartungswerte bei autokorrelierten Prozessen erhebliche Schwierigkeiten. Nun sind aber gerade die Wartezeiten sicher autokorrelierte Größen, da, falls der erste einer Warteschlange lange wartet, der zweite ebenfalls mindestens genauso lange warten muß. Unkorreliertes Verhalten haben wir nur für den Fall, daß die Wartezeiten Null sind, d.h. daß jeder Anruf sofort bedient wird.

Qualitativ gesprochen wird der Prozeß für unmittelbar hintereinander erfolgende Kranken- bzw. Notfallanrufe positiv korreliert sein, weswegen die geschätzte Standardabweichung gemäß (8-10) größer sein wird, als die durch (3-20) geschätzte. Wir wollen daher ein etwas vereinfachtes Verfahren für einen regenerativen Prozeß verwenden. Wir nehmen a priori an, daß weit auseinanderliegende Anrufe eine unkorrelierte Wartezeit aufweisen. Diese Regenerationszeitpunkte wollen wir aber nicht bestimmen, sondern als feste Zeitpunkte mit konstanten Zykluslängen annehmen (z.B. je 20 Notfallanrufe und je 30 Kranken-anrufe). Die Wartezeiten dieser 20 bzw. 30 aggregieren wir nun zu einer Einzig in Form des Mittelwertes und diese nehmen wir nun als die tatsächlichen und unabhängigen Beobachtungen an, für die wir Erwartungswert schätzen und die Standardabweichung gemäß (3-20). Betrachten wir die Situation genauer: Gegeben seien N Beobachtungen X_1, \dots, X_N . Diese aggregieren wir nun zu Blocks der Länge m , also

$$\begin{aligned}
 Y_1 &= \frac{X_1 + X_2 + \dots + X_m}{m} \\
 Y_2 &= \frac{X_{m+1} + \dots + X_{2m}}{m} \\
 &\dots\dots\dots \\
 Y_k &= \frac{X_{m(k-1)+1} + \dots + X_N}{m}
 \end{aligned}
 \tag{8-18}$$

wobei $k \cdot m = N$ gelten muß.

Hiebei sei angenommen, daß m so groß ist, daß gilt

$$\text{Cov}(Y_i, Y_j) = 0 \quad \text{für } i \neq j$$

Dann schätzen wir den Erwartungswert durch

$$\bar{X} = \frac{\sum_{i=1}^k Y_i}{k} = \frac{\sum_{i=1}^N X_i}{N}$$

und den dabei entstandenen Fehler durch

$$\text{Var}(\bar{X}) = \sigma^2 \sim S^2 = \frac{\sum_{i=1}^k (Y_i - \bar{X})^2}{k^2} = \frac{\sum_{i=1}^k Y_i^2}{k^2} - \frac{\bar{X}^2}{k} \tag{8-19}$$

Die Größe m (siehe 8-18) sollte dabei eine exogene Variable sein, damit man Sensitivitätsanalysen über m durchführen kann. Die angegebene Methode ist jedoch, darüber muß man sich im Klaren sein, nur eine heuristische Näherung, deren Vorteil darin besteht, daß das Schätzen der Kovarianzen oder das Bestimmen der Regenerationszeitpunkte nicht erforderlich ist.

8.7. Literatur:

M.A. Crone & D.L. Iglehart, Simulating Stable Stochastic Systems I-IV, Technical Reports, Control Analysis Corporation, Palo Alto, 1972-74.

D.Cox & P.Lewis, The statistical analysis of series of events, Wiley, New York, 1966.

8.8. Anhang: Computerprogramme

In FORTRAN

```
C PROGRAMM ZUR SIMULATION EINES EINFACHEN NOTFALLSYSTEMS
C WICHTIGE VERWENDETE GROESSEN:
C E(I) REGIONALE VERTEILUNG DER TELEFONANRUFEN
C F(I,J) MITTLERE FAHRZEITEN DER AMBULANZEN MIT UND OHNE BLAULICHT
C IN JEDE REGION
C G(I) ANKUNFT DER AMBULANZ I BEI DER EINSATZSTELLE NACH ERFOLG-
C TEM EINSATZ
C S(I) MITTLERE AUSLASTUNG DER AMBULANZ I.
C R(J,1) ANRUFZEITPUNKTE DER NOTFAELLE
C R(1,2) ANRUFZEITPUNKTE DER KRANKEN
C V(I),W(I) DIENT DER BERECHNUNG VON ERWARTUNGSWERT UND STANDARDAB-
C WEICHUNG DER WARTEZEITEN VON KRANKEN UND NOTFALL
C O ANZAHL NOTFALLANRUFEN PRO TAG
C P ANZAHL KRANKENANRUFEN PRO TAG
C K ANZAHL AMBULANZWAGEN
C T SIMULATIONSDAUER IN MINUTEN
C K1 ANZAHL WIEDERHOLUNGEN DER SIMULATION MIT VERSCHIEDENEN
C DATEN
C L ANZAHL ZU EINEM BLOCK ZUSAMMENGEFASSTER MESSUNGEN (WEGEN
C AUTOKORRELATION)
C AS BEGINN DER MESSUNGEN NACH STARTEN DER SIMULATION IN
C MINUTEN (WEGEN TRANSIENTEM ANFANGSZUSTAND)
C NX ANZAHL BEFOERDERTER KRANKEN
C ANZAHL BEFOERDERTER NOTFAELLE
C INTEGER AS
C DIMENSION E(12),F(12,2),G(15),S(15),R(1,2),V(4),W(4)

C EINLESEN UND DRUCKEN DER DATEN.

C READ 7,E
C READ 2,F
C PRINT 3
C PRINT 4
C DO 5 I=1,12
5 PRINT6,I,E(I),F(I,1),F(I,2)
C K2=0
C READ 1,K1
11 READ 1,K,T,O,P,Q,X
C READ 400,L,AS
C PRINT8,K
C PRINT9,O,P

C AUFBEREITEN DER DATEN

C X=RANF(X)
C A=-1.611
C B=-1.208
C C=2.819
C O=1440/O
C P=1440/P
C DO 15 I=1,11
15 E(I+1)=E(I+1)+E(I)
C DO 20 I=1,12
20 E(I)=E(I)/E(12)
C DO 25 I=1,K
C S(I)=0
```



```
25  G(I)=0
    R(1,1)=ZV2(O)
    R(1,2)=ZV2(P)
    DO 26 I=1,4
    V(I)=0
26  W(I)=0
    NX=0
    N=0
    J=1
    M1=0
    M2=0
    U1=0
    U2=0

C    ABFRAGE, OB SIMULATIONSDAUER T ERREICHT

30  IF(G(J) .GT. T) GOTO 300
C    ENTSCHEIDUNG OB KRANKER ODER NOTFALL ABGEHOLT WIRD
31  IF(R(1,1)-R(1,2)) 40,40,45
40  DO 35 I=1,K
    IF(R(1,1)-G(I)) 35,50,50
50  J=I
    GOTO80
35  CONTINUE

C    ZUORDNUNG EINER AMBULANZ ZUM NOTFALL

75  H=G(1)
    J=1
    DO 55 I=2,K
    IF(G(I)-H) 60,55,55
60  H=G(I)
    J=I
55  CONTINUE
80  I=NV(E)
    Z1=ZV1(A,B,C)
    Z2=ZV1(A,B,C)
    Z1=Z1*F(I,2)/Q+F(I,2)
    Z2=Z2*F(I,2)/Q+F(I,2)
    H1=G(J)-R(1,1)
    G(J)=AMAX1(R(1,1),G(J))+Z1+Z2
    R(1,1)=ZV2(O)+R(1,1)
    IF(G(J) .LT. AS) GOTO 30

C    BERECHNEN VON STATISTIKEN

    H1=AMAX1(H1,0.)
    S(J)=S(J)+Z1+Z2
    U1=U1+H1
    M1=M1+1
    IF(M1 .LT. L) GOTO 30
    U1=U1/M1
    W(1)=W(1)+U1
    V(1)=V(1)+U1*U1
    U1=0
    M1=0
    N=N+1
```

```

GOTO 30

C   FESTSTELLEN, OB EINE AMBULANZ FREI IST, ODER OB KRANKER WARTEN MUSS
45  DO 100 I=1,K
    IF(R(1,2)-G(I)) 100,105,105
105  J=I
    GOTO 110
100  CONTINUE
C   KRANKER MUSS WARTEN
    H=G(1)
    J=1
    DO 115 I=2,K
    IF(G(I)-H) 120,115,115
120  H=G(I)
    J=I
115  CONTINUE
C   FALLS INZWISCHEN NOTFALL ANRUFT, WIRD DIESER ZUERST BEDIENT
    IF(H-R(1,1)) 110,80,80

C   AMBULANZ HOLT KRANKEN AB

110  I=NV(E)
    Z1=ZV1(A,B,C)
    Z2=ZV1(A,B,C)
    Z1=Z1*F(I,1)/Q+F(I,1)
    Z2=Z2*F(I,1)/Q+F(I,1)
    H1=G(J)-R(1,2)
    G(J)=AMAX1(R(1,2),G(J))+Z1+Z2
    R(1,2)=ZV2(P)+R(1,2)
    IF(G(J) .LT. AS) GOTO 30
C   BERECHNEN VON STATISTIKEN
    S(J)=S(J)+Z1+Z2
    H1=AMAX1(H1,0.)
    U2=U2+H1
    M2=M2+1
    IF(M2 .LT. L) GOTO 30
    U2=U2/M2
    W(4)=W(4)+U2
    V(4)=V(4)+U2*U2
    NX=NX+1
    U2=0
    M2=0
    GOTO 30

C   BERECHNEN UND AUSDRUCKEN DER RESULTATE

300  CONTINUE
    DO 305 I=1,K
    G(I)=AMAX1(T,G(I))-AS
305  S(I)=S(I)/G(I)
    W(1)=W(1)/N
    W(4)=W(4)/NX
    V(4)=SQRT(V(4)-NX*W(4)*W(4))/NX
    V(1)=SQRT(V(1)-N *W(1)*W(1))/N
    PRINT 10,T
    PRINT 320

```

```

PRINT 326,W(1),V(1)
PRINT 329,W(4),V(4)
DO 335 I=1,K
335 PRINT 325,I,S(I)
K2=K2+1
N =N *L*1440./(T-AS)
NX=NX*L*1440./(T-AS)
PRINT 401,N,NX
PRINT 410,AS,L
IF(K2 .LT. K1) GOTO 11
1 FORMAT(I2,7(F11.1))
2 FORMAT(24(F3.1))
3 FORMAT(17H1DIE DATEN LAUTEN)
4 FORMAT(6HOKREIS,5X,9HEINWOHNER,10X,10HFAHRZEITEN)
6 FORMAT(1H ,I4,F13.0,2F14.1)
7 FORMAT(12(F6.0))
8 FORMAT(21H1ANZAHL KRANKENWAGEN=,I3)
9 FORMAT(9HOES RUFEN,F4.0,' NOTFAELLE UND ',F4.0,28H KRANKE PRO TAG
*IM MITTEL AN)
10 FORMAT(1H0,F11.0,25H MINUTEN WERDEN SIMULIERT)
320 FORMAT(1H0,31X,' ERWARTUNGSWERT STANDARDABWEICHUNG')
325 FORMAT(15H AUSLASTUNG DER,I2,13H-TEN AMBULANZ,2F14.4)
326 FORMAT(30H AMBULANZ FUER NOTFALL FREI ,2F14.4)
329 FORMAT(' AMBULANZ FUER KRANKEN FREI ',2F14.4)
400 FORMAT(I3,I5)
401 FORMAT(' OPRO TAG WURDEN',I3,' NOTFAELLE UND',I4,' KRANKE BEFOERDER
*T')
410 FORMAT(' OGEMESSEN WURDE AB',I5,' MINUTEN IN BLOECKEN ZU JE',I3)

```

```

FUNCTION ZV1(A,B,C)
C SUBROUTINE ZUM ERZEUGEN DER ZUFAELLIGEN FAHRZEITEN
D=B-A
EX=C-B
Z=RANF(X )*(D+EX)
IF(Z-D) 1,1,2
1 ZV1=SQRT(Z*D)+A
RETURN
2 ZV1=C-SQRT((Z-D)*EX)
RETURN

```

```

FUNCTION ZV2(A)
C SUBROUTINE ZUM ERZEUGEN DER ZUFAELLIGEN ANRUFZEITEN
ZV2=-A*LOG(RANF(X ))*A
RETURN

```

```

FUNCTION NV(A)
C SUBROUTINE ZUM ERZEUGEN DER ZUFAELLIGEN ORTE DER ANRUFER
DIMENSION A(12)
Z=RANF(X )
DO 1 IX=1,12
IF(Z-A(IX)) 2,2,1
1 CONTINUE
2 NV=IX
RETURN

```

```

C  FUNCTION RANF(X)
    SUBROUTINE ZUM ERZEUGEN VON IN (0,1) GLEICHVERTEILTEN ZUFALLSZAHLEN.
    IR=X*2.**35.
    IR=NRAND(IR)
    X=IR/2.**35.
    RANF=X
    RETURN
    END
  
```

DIE DATEN LAUTEN

KREIS	EINWOHNER	FAHRZEITEN	
1	9000.	14.0	10.0
2	32600.	21.0	15.0
3	51700.	17.0	12.0
4	31800.	15.0	11.0
5	12500.	15.0	11.0
6	37800.	14.0	10.5
7	39100.	18.5	13.5
8	20300.	17.0	12.5
9	47200.	22.0	15.0
10	35800.	20.0	14.0
11	56900.	21.5	14.5
12	33000.	21.0	15.0

ANZAHL KRANKENWAGEN= 15

ES RUFEN 14. NOTFAELLE UND 100. KRANKE PRO TAG IM MITTEL AN
144000. MINUTEN WERDEN SIMULIERT

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	.0000	.0000
AMBULANZ FUER KRANKEN FREI	.0000	.0000
AUSLASTUNG DER 1-TEN AMBULANZ	.7447	
AUSLASTUNG DER 2-TEN AMBULANZ	.6520	
AUSLASTUNG DER 3-TEN AMBULANZ	.5347	
AUSLASTUNG DER 4-TEN AMBULANZ	.4114	
AUSLASTUNG DER 5-TEN AMBULANZ	.2801	
AUSLASTUNG DER 6-TEN AMBULANZ	.1735	
AUSLASTUNG DER 7-TEN AMBULANZ	.0836	
AUSLASTUNG DER 8-TEN AMBULANZ	.0416	
AUSLASTUNG DER 9-TEN AMBULANZ	.0169	
AUSLASTUNG DER 10-TEN AMBULANZ	.0027	
AUSLASTUNG DER 11-TEN AMBULANZ	.0012	
AUSLASTUNG DER 12-TEN AMBULANZ	.0003	
AUSLASTUNG DER 13-TEN AMBULANZ	.0000	
AUSLASTUNG DER 14-TEN AMBULANZ	.0000	
AUSLASTUNG DER 15-TEN AMBULANZ	.0000	

PRO TAG WURDEN 13 NOTFAELLE UND 103 KRANKE BEFOERDERT
GEMESSEN WURDE AB 1440 MINUTEN IN BLOECKEN ZU JE 20
ANZAHL KRANKENWAGEN= 12

ES RUFEN 14. NOTFAELLE UND 100. KRANKE PRO TAG IM MITTEL AN
144000. MINUTEN WERDEN SIMULIERT

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	.0000	.0000
AMBULANZ FUER KRANKEN FREI	.0001	.0001
AUSLASTUNG DER 1-TEN AMBULANZ	.7534	
AUSLASTUNG DER 2-TEN AMBULANZ	.6529	
AUSLASTUNG DER 3-TEN AMBULANZ	.5522	

AUSLASTUNG DER 4-TEN AMBULANZ	.4227
AUSLASTUNG DER 5-TEN AMBULANZ	.2912
AUSLASTUNG DER 6-TEN AMBULANZ	.1795
AUSLASTUNG DER 7-TEN AMBULANZ	.0940
AUSLASTUNG DER 8-TEN AMBULANZ	.0397
AUSLASTUNG DER 9-TEN AMBULANZ	.0171
AUSLASTUNG DER 10-TEN AMBULANZ	.0042
AUSLASTUNG DER 11-TEN AMBULANZ	.0019
AUSLASTUNG DER 12-TEN AMBULANZ	.0002

PRO TAG WURDEN 13 NOTFAELLE UND 103 KRANKE BEFOERDERT
 GEMESSEN WURDE AB 1440 MINUTEN IN BLOECKEN ZU JE 20
 ANZAHL KRANKENWAGEN= 10

ES RUFEN 14. NOTFAELLE UND 100. KRANKE PRO TAG IM MITTEL AN
 144000. MINUTEN WERDEN SIMULIERT

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	.0035	.0035
AMBULANZ FUER KRANKEN FREI	.0048	.0020
AUSLASTUNG DER 1-TEN AMBULANZ	.7604	
AUSLASTUNG DER 2-TEN AMBULANZ	.6723	
AUSLASTUNG DER 3-TEN AMBULANZ	.5622	
AUSLASTUNG DER 4-TEN AMBULANZ	.4288	
AUSLASTUNG DER 5-TEN AMBULANZ	.3061	
AUSLASTUNG DER 6-TEN AMBULANZ	.1924	
AUSLASTUNG DER 7-TEN AMBULANZ	.1024	
AUSLASTUNG DER 8-TEN AMBULANZ	.0489	
AUSLASTUNG DER 9-TEN AMBULANZ	.0191	
AUSLASTUNG DER 10-TEN AMBULANZ	.0056	

PRO TAG WURDEN 13 NOTFAELLE UND 103 KRANKE BEFOERDERT
 GEMESSEN WURDE AB 1440 MINUTEN IN BLOECKEN ZU JE 20
 ANZAHL KRANKENWAGEN= 8

ES RUFEN 14. NOTFAELLE UND 100. KRANKE PRO TAG IM MITTEL AN
 144000. MINUTEN WERDEN SIMULIERT

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	.0752	.0213
AMBULANZ FUER KRANKEN FREI	.0888	.0154
AUSLASTUNG DER 1-TEN AMBULANZ	.7678	
AUSLASTUNG DER 2-TEN AMBULANZ	.6806	
AUSLASTUNG DER 3-TEN AMBULANZ	.5649	
AUSLASTUNG DER 4-TEN AMBULANZ	.4491	
AUSLASTUNG DER 5-TEN AMBULANZ	.3288	
AUSLASTUNG DER 6-TEN AMBULANZ	.2030	
AUSLASTUNG DER 7-TEN AMBULANZ	.1118	
AUSLASTUNG DER 8-TEN AMBULANZ	.0564	

PRO TAG WURDEN 13 NOTFAELLE UND 103 KRANKE BEFOERDERT
 GEMESSEN WURDE AB 1440 MINUTEN IN BLOECKEN ZU JE 20
 ANZAHL KRANKENWAGEN= 6

ES RUFEN 14. NOTFAELLE UND 100. KRANKE PRO TAG IM MITTEL AN
 144000. MINUTEN WERDEN SIMULIERT

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	.6481	.0592
AMBULANZ FUER KRANKEN FREI	1.3741	.1109
AUSLASTUNG DER 1-TEN AMBULANZ	.7803	
AUSLASTUNG DER 2-TEN AMBULANZ	.7002	
AUSLASTUNG DER 3-TEN AMBULANZ	.6041	
AUSLASTUNG DER 4-TEN AMBULANZ	.4875	
AUSLASTUNG DER 5-TEN AMBULANZ	.3787	
AUSLASTUNG DER 6-TEN AMBULANZ	.2528	

PRO TAG WURDEN 14 NOTFAELLE UND 103 KRANKE BEFOERDERT
GEMESSEN WURDE AB 1440 MINUTEN IN BLOECKEN ZU JE 20
ANZAHL KRANKENWAGEN= 4
ES RUFEN 14. NOTFAELLE UND 100. KRANKE PRO TAG IM MITTEL AN
144000. MINUTEN WERDEN SIMULIERT

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	5.0925	.2173
AMBULANZ FUER KRANKEN FREI	20.3054	1.0426
AUSLASTUNG DER 1-TEN AMBULANZ	.8827	
AUSLASTUNG DER 2-TEN AMBULANZ	.8396	
AUSLASTUNG DER 3-TEN AMBULANZ	.7831	
AUSLASTUNG DER 4-TEN AMBULANZ	.7160	

PRO TAG WURDEN 15 NOTFAELLE UND 102 KRANKE BEFOERDERT
GEMESSEN WURDE AB 1440 MINUTEN IN BLOECKEN ZU JE 20
ANZAHL KRANKENWAGEN= 3
ES RUFEN 14. NOTFAELLE UND 100. KRANKE PRO TAG IM MITTEL AN
144000. MINUTEN WERDEN SIMULIERT

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	11.2544	.2526
AMBULANZ FUER KRANKEN FREI	6001.1013	157.1469
AUSLASTUNG DER 1-TEN AMBULANZ	1.0000	
AUSLASTUNG DER 2-TEN AMBULANZ	.9998	
AUSLASTUNG DER 3-TEN AMBULANZ	1.0000	

PRO TAG WURDEN 14 NOTFAELLE UND 93 KRANKE BEFOERDERT
GEMESSEN WURDE AB 1440 MINUTEN IN BLOECKEN ZU JE 20
ANZAHL KRANKENWAGEN= 2
ES RUFEN 14. NOTFAELLE UND 100. KRANKE PRO TAG IM MITTEL AN
144000. MINUTEN WERDEN SIMULIERT

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	15.3900	.3507
AMBULANZ FUER KRANKEN FREI	30651.8706	1026.2977
AUSLASTUNG DER 1-TEN AMBULANZ	1.0001	
AUSLASTUNG DER 2-TEN AMBULANZ	1.0002	

PRO TAG WURDEN 14 NOTFAELLE UND 58 KRANKE BEFOERDERT
GEMESSEN WURDE AB 1440 MINUTEN IN BLOECKEN ZU JE 20

EXOGENOUS EVENT START

EREIGNIS, WELCHES DIE SIMULATION IMITIERT

CREATE NTANR

CREATE KRANR

DIE EREIGNISSE NOTFALLANRUF UND KRANKENANRUF WERDEN AUFGERUFEN

CAUSE NTANR AT TIME

CAUSE KRANR AT TIME

DIE AMBULANZWAGEN WERDEN 'ERZEUGT'

DO, FOR J= (1) (ANZ)

CREATE AMBUL

LET S(AMBUL)=0

LET N(AMBUL)=J

AMBULANZWAGEN SIND ZU BEGINN ALLE IN DER EINSATZSTELLE

FILE AMBUL IN SPIT

REPEAT

RETURN

END

ENDOGENOUS EVENT NTANR

NOTFALL RUFT AN

CREATE NOTFA

ZEIT DES ANRUFES UND ORT BES ANRUFENDEN WERDEN NOTIERT

LET TIM(NOTFA)=TIME

LET ORT(NOTFA)=I

FILE NOTFA IN NWAR

CREATE AMBAB

EREIGNIS AMBULANZABFAHRT WIRD AUFGERUFEN

CAUSE AMBAB AT TIME

ZEITPUNKT DES NAECHSTEN NOTFALLANRUFES WIRD BESTIMMT

CAUSE NTANR AT TIME - ALOG(RANDM)/14.

RETURN

END

ENDOGENOUS EVENT KRANR

KRANKER RUFT AN

CREATE KRANK

ZEIT DES ANRUFES UND ORT DES ANRUFENDEN WERDEN NOTIERT

LET TIM(KRANK)=TIME

LET ORT(KRANK)=I

FILE KRANK IN KWAR

CREATE AMBAB

EREIGNIS AMBULANZABFAHRT WIRD AUFGERUFEN

CAUSE AMBAB AT TIME

ZEITPUNKT DES NAECHSTEN KRANKENANRUFES WIRD BESTIMMT

CAUSE KRANR AT TIME-ALOG(RANDM)/100.

RETURN

END

ENDOGENOUS EVENT AMBAB

DESTROY AMBAB

IST EINE AMBULZNA FREI?

IF SPIT IS EMPTY, RETURN

IST EIN NOTFALL IN DER WARTESCHLANGE

IF NWAR IS EMPTY, GO TO 100

REMOVE FIRST NOTFA FROM NWAR

REMOVE FIRST AMBUL FROM SPIT

BERECHNEN VON STATISTIKEN

```

LET T = B(ORT(NOTFA)) * (U/4. +1.)
LET T= T + B(ORT(NOTFA)) * (U/4.+1.)
IF TIME LE AS, GO TO 10
LET S(AMBUL)=S(AMBUL)+T
LET E3=E3+TIME-TIM(NOTFA)
LET V3=V3+1.
IF V3 LT L, GO TO 10
LET E3=E3/V3
LET E1=E1+E3
LET V1=V1+E3*E3
LET N1=N1+1
LET E3=0
LET V3=0
10 LET ZEIT(AMBUL) = TIME + T
   DESTROY NOTFA
C   AMBULANZ FAEHRT AB
   FILE AMBUL IN FAHR
   CREATE AMBAN
C   FESTLEGEN DES RUECKKEHRZEITPUNKTES
   CAUSE AMBAN AT TIME + T
   RETURN
C   IST EIN KRANKER IN DER WARTESCHLANGE
100 IF KWAR IS EMPTY, RETURN
   REMOVE FIRST KRANK FROM KWAR
   REMOVE FIRST AMBUL FROM SPIT
C   BERECHNEN VON STATISTIKEN
   LET T= OB(ORT(KRANK))*(U/4.+1.)
   LET T= OB(ORT(KRANK))*(U/4.+1.)+T
   IF TIME LE AS, GO TO 20
   LET S(AMBUL)=S(AMBUL)+T
   LET E4=E4+TIME-TIM(KRANK)
   LET V4=V4+1.
   IF V4 LT L, GO TO 20
   LET E4=E4/V4
   LET E2=E2+E4
   LET V2=V2+E4*E4
   LET E4=0
   LET V4=0
   LET N2=N2+1
20 LET ZEIT(AMBUL)=TIME + T
   DESTROY KRANK
C   AMBULANZ FAEHRT AB
   FILE AMBUL IN FAHR
   CREATE AMBAN
C   FESTLEGEN DES RUECKKEHRZEITPUNKTES
   CAUSE AMBAN AT TIME +T
   RETURN
END

ENDOGENOUS EVENT AMBAN
C   AMBULANZ KEHRT ZUR EINSATZSTELLE ZURUECK
   DESTROY AMBAN
   CREATE AMBAB
C   UEBERPRUEFEN, OB AMBULANZ GLEICH WIEDER ABFAHREN MUSS
   CAUSE AMBAB AT TIME
   REMOVE FIRST AMBUL FROM FAHR
   FILE AMBUL IN SPIT

```

RETURN
END

EXOGENOUS EVENT END

SIMULATION WIRD TERMINIERT

BERECHNEN UND DRUCKEN DER RESULTATE

WRITE ON 6, TIME

FORMAT('1',M5.2.2,' TAGE WERDEN SIMULIERT')

WRITE ON 6, ANZ

FORMAT('OANZAHL KRANKENWAGEN =',I3)

LET IN1=IFIX(FLOAT(N1)*L/(TIME-AS))

LET IN2=IFIX(FLOAT(N2)*L/(TIME-AS))

WRITE ON 6, IN1,IN2

FORMAT('OES WURDEN',I3,' NOTFAELLE UND',I4,' KRANKE PRO TAG IM MIT
XTEL BEFOERDERT')

WRITE ON 6

FORMAT('OES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN
X')

WRITE ON 6

FORMAT('0',S31,'ERWARTUNGSWERT STANDARDABWEICHUNG')

LET E1=E1/FLOAT(N1)

LET E2=E2/FLOAT(N2)

LET V1=SQRT(V1-FLOAT(N1)*E1*E1)/FLOAT(N1)

LET V2=SQRT(V2-FLOAT(N2)*E2*E2)/FLOAT(N2)

LET E1=E1*1440.

LET E2=E2*1440.

LET V1=V1*1440.

LET V2=V2*1440.

WRITE ON 6, E1,V1

FORMAT(' AMBULANZ FUER NOTFALL FREI ',2D10.4)

WRITE ON 6, E2,V2

FORMAT(' AMBULANZ FUER KRANKEN FREI ',2D10.4)

10 IF SPIT IS EMPTY, GO TO 20

REMOVE FIRST AMBUL FROM SPIT

GO TO 30

20 IF FAHR IS EMPTY, GO TO 40

REMOVE FIRST AMBUL FROM FAHR

30 LET S(AMBUL)=S(AMBUL)/(TIME-AS)

WRITE ON 6, N(AMBUL),S(AMBUL)

FORMAT(' AUSLASTUNG DER',I2,'-TEN. AMBULANZ',D10.4)

DESTROY AMBUL

GO TO 10

40 STOP

END

FUNCTION U

FUNKTION ZUM BESTIMMEN DER ZUFAELLIGEN FAHRZEITEN

LET A = -1.611

LET F = -1.208

LET C = 2.819

LET D = F-A

LET E = C-F

LET Z = RANDM * (D+E)

IF (Z-D) 1,1,2

1 LET U = SQRT(Z+D) + A

RETURN

2 LET U = C - SQRT((Z-D) * E)

RETURN

END

100.00.00 TAGE WERDEN SIMULIERT

ANZAHL KRANKENWAGEN = 15

ES WURDEN 13 NOTFAELLE UND 101 KRANKE PRO TAG IM MITTEL BEFOERDERT

ES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	0.	0.
AMBULANZ FUER KRANKEN FREI	0.	0.
AUSLASTUNG DER 13-TEN AMBULANZ	0.1917	
AUSLASTUNG DER 12-TEN AMBULANZ	0.1914	
AUSLASTUNG DER 15-TEN AMBULANZ	0.1965	
AUSLASTUNG DER 10-TEN AMBULANZ	0.1913	
AUSLASTUNG DER 5-TEN AMBULANZ	0.1931	
AUSLASTUNG DER 3-TEN AMBULANZ	0.1915	
AUSLASTUNG DER 11-TEN AMBULANZ	0.1939	
AUSLASTUNG DER 8-TEN AMBULANZ	0.1932	
AUSLASTUNG DER 9-TEN AMBULANZ	0.1943	
AUSLASTUNG DER 1-TEN AMBULANZ	0.1937	
AUSLASTUNG DER 2-TEN AMBULANZ	0.1930	
AUSLASTUNG DER 4-TEN AMBULANZ	0.1936	
AUSLASTUNG DER 14-TEN AMBULANZ	0.1938	
AUSLASTUNG DER 6-TEN AMBULANZ	0.1934	
AUSLASTUNG DER 7-TEN AMBULANZ	0.1938	

100.00.00 TAGE WERDEN SIMULIERT

ANZAHL KRANKENWAGEN = 12

ES WURDEN 13 NOTFAELLE UND 101 KRANKE PRO TAG IM MITTEL BEFOERDERT

ES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	0.	0.
AMBULANZ FUER KRANKEN FREI	0.0008	0.0008
AUSLASTUNG DER 8-TEN AMBULANZ	0.2434	
AUSLASTUNG DER 3-TEN AMBULANZ	0.2399	
AUSLASTUNG DER 6-TEN AMBULANZ	0.2406	
AUSLASTUNG DER 5-TEN AMBULANZ	0.2418	
AUSLASTUNG DER 12-TEN AMBULANZ	0.2419	
AUSLASTUNG DER 11-TEN AMBULANZ	0.2409	
AUSLASTUNG DER 2-TEN AMBULANZ	0.2427	
AUSLASTUNG DER 7-TEN AMBULANZ	0.2402	

AUSLASTUNG DER 1-TEN AMBULANZ	0.2393
AUSLASTUNG DER 10-TEN AMBULANZ	0.2428
AUSLASTUNG DER 9-TEN AMBULANZ	0.2391
AUSLASTUNG DER 4-TEN AMBULANZ	0.2454

100.00.00 TAGE WERDEN SIMULIERT

ANZAHL KRANKENWAGEN = 10

ES WURDEN 13 NOTFAELLE UND 101 KRANKE PRO TAG IM MITTEL BEFOERDERT

ES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	0.	0.
AMBULANZ FUER KRANKEN FREI	0.0024	0.0021
AUSLASTUNG DER 1-TEN AMBULANZ	0.2874	
AUSLASTUNG DER 4-TEN AMBULANZ	0.2920	
AUSLASTUNG DER 5-TEN AMBULANZ	0.2891	
AUSLASTUNG DER 8-TEN AMBULANZ	0.2906	
AUSLASTUNG DER 6-TEN AMBULANZ	0.2885	
AUSLASTUNG DER 7-TEN AMBULANZ	0.2919	
AUSLASTUNG DER 3-TEN AMBULANZ	0.2885	
AUSLASTUNG DER 9-TEN AMBULANZ	0.2900	
AUSLASTUNG DER 10-TEN AMBULANZ	0.2876	
AUSLASTUNG DER 2-TEN AMBULANZ	0.2911	

100.00.00 TAGE WERDEN SIMULIERT

ANZAHL KRANKENWAGEN = 8

ES WURDEN 13 NOTFAELLE UND 101 KRANKE PRO TAG IM MITTEL BEFOERDERT

ES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	0.0427	0.0160
AMBULANZ FUER KRANKEN FREI	0.0488	0.0112
AUSLASTUNG DER 5-TEN AMBULANZ	0.3645	
AUSLASTUNG DER 6-TEN AMBULANZ	0.3625	
AUSLASTUNG DER 2-TEN AMBULANZ	0.3603	
AUSLASTUNG DER 3-TEN AMBULANZ	0.3609	
AUSLASTUNG DER 1-TEN AMBULANZ	0.3612	
AUSLASTUNG DER 4-TEN AMBULANZ	0.3625	
AUSLASTUNG DER 7-TEN AMBULANZ	0.3612	
AUSLASTUNG DER 8-TEN AMBULANZ	0.3623	

100.00.00 TAGE WERDEN SIMULIERT

ANZAHL KRANKENWAGEN = 6

ES WURDEN 13 NOTFAELLE UND 101 KRANKE PRO TAG IM MITTEL BEFOERDERT

ES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	0.3433	0.0425
AMBULANZ FUER KRANKEN FREI	0.6731	0.0690
AUSLASTUNG DER 3-TEN AMBULANZ	0.4802	
AUSLASTUNG DER 6-TEN AMBULANZ	0.4862	
AUSLASTUNG DER 4-TEN AMBULANZ	0.4862	
AUSLASTUNG DER 2-TEN AMBULANZ	0.4807	
AUSLASTUNG DER 5-TEN AMBULANZ	0.4852	
AUSLASTUNG DER 1-TEN AMBULANZ	0.4777	

100.00.00 TAGE WERDEN SIMULIERT
ANZAHL KRANKENWAGEN = 4
ES WURDEN 13 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL BEFOERDERT
ES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	3.0258	0.1849
AMBULANZ FUER KRANKEN FREI	8.4022	0.4279
AUSLASTUNG DER 2-TEN AMBULANZ	0.7226	
AUSLASTUNG DER 1-TEN AMBULANZ	0.7210	
AUSLASTUNG DER 3-TEN AMBULANZ	0.7210	
AUSLASTUNG DER 4-TEN AMBULANZ	0.7174	

100.00.00 TAGE WERDEN SIMULIERT
ANZAHL KRANKENWAGEN = 3
ES WURDEN 13 NOTFAELLE UND 101 KRANKE PRO TAG IM MITTEL BEFOERDERT
ES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	9.2095	0.2601
AMBULANZ FUER KRANKEN FREI	163.7918	6.3623
AUSLASTUNG DER 2-TEN AMBULANZ	0.9707	
AUSLASTUNG DER 1-TEN AMBULANZ	0.9675	
AUSLASTUNG DER 3-TEN AMBULANZ	0.9671	

100.00.00 TAGE WERDEN SIMULIERT
ANZAHL KRANKENWAGEN = 2
ES WURDEN 13 NOTFAELLE UND 66 KRANKE PRO TAG IM MITTEL BEFOERDERT
ES RUFEN 14 NOTFAELLE UND 100 KRANKE PRO TAG IM MITTEL AN

	ERWARTUNGSWERT	STANDARDABWEICHUNG
AMBULANZ FUER NOTFALL FREI	13.5804	0.2522
AMBULANZ FUER KRANKEN FREI	24750.0527	778.0929
AUSLASTUNG DER 1-TEN AMBULANZ	0.9994	
AUSLASTUNG DER 2-TEN AMBULANZ	1.0006	

9. ÜBERGANG ZU KONTINUIERLICHEN MODELLEN

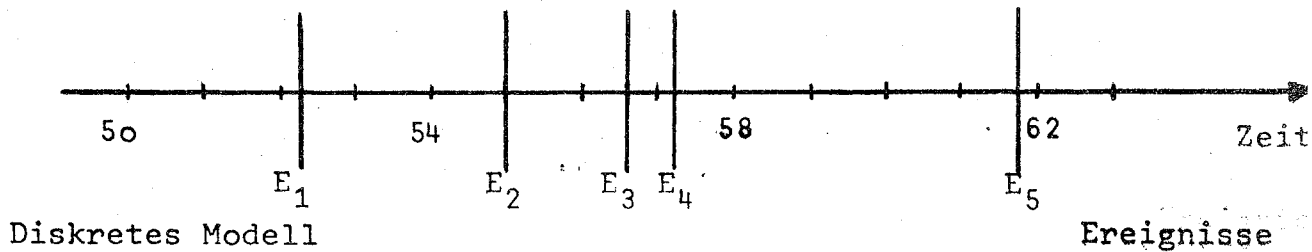
9.1. Arten des Zeitfortschrittes

Bislang haben wir uns nur mit Modellen beschäftigt, bei denen der Zeitfortschritt ereignisabhängig war und daher variabel (sogenannte diskrete Modelle). Bestimmte Systeme können aber auf diese Art nicht simuliert werden und man muß dann den Zeitfortschritt als konstant annehmen, d.h. die Zeit springt immer um einen konstanten Betrag vorwärts, um eine Einheitszeit. Man unterscheidet also zwei Arten von Zeitfortschritt: Von Ereignis zu Ereignis oder um eine Zeiteinheit.

Die Programmierung beim Fortschritt um Zeiteinheiten ist wesentlich einfacher, da keine Ereignisliste geführt werden muß. Modelle dieser Art dienen zur Beschreibung kontinuierlicher Prozesse, also solcher, in denen sich der Zustand des Systems dauernd verändert. In diesem Fall kann man nicht mehr von Ereignissen sprechen. Man nennt diese Modelle daher auch kontinuierliche Modelle und die Modelle mit ereignisorientierten Zeitfortschritt diskret. Kontinuierliche Modelle beinhalten also Zustandsgrößen, welche sich in jedem Zeitpunkt ändern, und man approximiert ein solches System, indem man den Zustand in festen Zeitabschnitten verändert. Ein typisches Beispiel wären also Differenzgleichungen zur Approximation von Differentialgleichungen, wobei allerdings stochastische Elemente durchaus zugelassen sind.

Allerdings können kontinuierliche Modelle auch zur Simulation diskreter Modelle herangezogen werden, indem die zu diskreten Zeitpunkten sich ändernden Zustandsvariablen umdefiniert werden zu kontinuierlichen, wobei sie stets über verschieden lange Intervalle konstant sind. Dies ist in Figur 9-1 dargestellt.

Kontinuierliches Modell



Figur 9-1

Zwei Arten von Zeitfortschritt

Da ein diskretes Modell durch ein kontinuierliches ersetzt werden kann, stellt sich die Frage, wann ein solches Vorgehen sinnvoll ist. Aus Figur 9-1 erkennt man zunächst, daß ein kontinuierliches Modell ein diskretes nur approximieren kann, da der genaue Zeitpunkt des Ereignisses nicht festgestellt wird, sondern nur das Zeitintervall, in dem das Ereignis passiert. Ist also Δt die Länge eines Zeitintervalls, so begeht man maximal einen Fehler beim Festlegen des Zeitpunktes eines Ereignisses von $\Delta t - \epsilon$, $\epsilon > 0$.

Betrachten wir nun folgendes hypothetisches Simulationsmodell:

- i) Der Zustand des Systems kann zu jedem Zeitpunkt durch k Variable beschrieben werden
- ii) Jede Variable bleibt im Mittel während n Zeitperioden Δt konstant
- iii) Das Simulationsmodell soll für N Zeitperioden durchgerechnet werden.

In jedem Intervall muß der Zustand aller k Variablen festgestellt werden. In N Zeitperioden sind dies also $N \cdot k$ Abfragen. Um die Anzahl Abfragen bei ereignisorientierter Simulation festzustellen, muß man die erwartete Anzahl Ereignisse im gesamten Run berechnen. Wir nehmen hiefür an, daß jede Variable unabhängig von den anderen ist. Da jede Variable während n Zeitperioden konstant

bleibt, sind in N Perioden N/m Ereignisse pro Variablen, also insgesamt $k \cdot N/m$ Ereignisse zu erwarten. Bei jedem Ereignis müssen wieder alle Variablen gecheckt werden, wofür wiederum k Abfragen nötig sind, also insgesamt $k^2 N/m$ Abfragen. Daher wird die ereignisorientierte Simulation schneller sein, falls

$$\begin{aligned} N \cdot k &> N \cdot k^2 / m \\ \Rightarrow 1 &> k / m \end{aligned} \quad (9-1)$$

Die Aussage in (9-1) ist im Grunde trivial. Sie sagt nichts anderes, als daß diskrete Simulation besser ist, falls im Mittel pro Zeitintervall weniger als ein Ereignis zu erwarten ist (siehe Figur 9-1). Ist es mehr als ein Ereignis pro Intervall, so ist das kontinuierliche Modell effizienter, da dann alle Ereignisse pro Intervall zu einem einzigen aggregiert werden. Die Wahl des Zeitintervalls Δt ist also offensichtlich äußerst kritisch, einerseits für die Güte der Approximation eines diskreten Modells, andererseits für die Effizienz, wobei gilt, je effizienter, desto größer muß Δt sein, desto schlechter wird die Approximation.

9.2. Wahl der Zeiteinheit

Das Problem der Wahl der Zeiteinheit ist vor allem im Zusammenhang mit dem numerischen Lösen von Differentialgleichungen genauer untersucht worden. Allerdings sind bei Simulationsmodellen die Zustandsvariablen stochastisch und die Beziehungen zwischen den Variablen äußerst komplex. Daher erscheint es nicht möglich, genaue Kriterien für die Länge der Zeiteinheit anzugeben. Um es nochmals zusammenzufassen: Die Wahl der Zeiteinheit beeinflusst

- Numerische Fehler: Falls das Intervall klein, daher bei gegebener Simulationsdauer sehr viele Intervallsschritte durchgeführt werden, können Rundungsfehler das Resultat verfälschen.

- Diskretisierungsfehler: Bei großen Intervallen werden Fehler bei der Veränderung von Variablen entstehen (siehe Figur 9-1).
- Effizienz: Kleine Intervalle bewirken rechenzeitintensive Simulationsruns.

Insbesondere in der Ökonomie wird die Zeiteinheit primär durch die Datensituation festgelegt. Bei vierteljährlich erhobenen Daten, erscheint es nicht sinnvoll kürzere Zeiteinheiten zu wählen.

So bleibt als wichtigstes Kriterium zur Beurteilung von gewählten Zeiteinheiten die nachträgliche Sensitivitätsanalyse. Es muß also untersucht werden, wie die Resultate durch die Veränderung der Zeiteinheit variieren.

9.3. DYNAMO

Es soll nun die Simulationssprache DYNAMO als Beispiel einer Sprache für kontinuierliche Modelle besprochen werden. DYNAMO löst Probleme der folgenden Art:

$$y_{n-1} = k(x_{n-1}, y_{n-2}, t_{n-1}) \quad (9-2)$$

$$x_n = x_{n-1} + \Delta t \cdot m(y_{n-1}) \quad (9-3)$$

$$\Delta t = t_n - t_{n-1} = \text{const} \quad (9-4)$$

$$y_n \in \mathbb{R}^S, \quad x_n \in \mathbb{R}^P$$

x_n wird auch Level und y_n Rate genannt. (9-2) beschreibt also ein Differenzengleichungssystem 1. Ordnung. (9-3) kann auch als einfache numerische Integration für das Differentialgleichungssystem

$$\frac{dx}{dt} = m(y) \quad (9-5)$$

angesehen werden. Der Anwendungsbereich solcher Systeme soll hier noch nicht genauer erörtert werden. Hiefür verweisen wir auf Kapitel 10. Wichtig ist vor allem (9-2). Die Funktion y_n

ist nämlich nicht explizit als Funktion des Levels x_n und der Zeit t_n definiert, sondern als rekursive Funktion, welche auch von y_{n-1} abhängt. Dadurch erweitert sich der Bereich möglicher Funktionen stark. Die Funktion $k(.)$ muß nicht deterministisch, sie kann auch stochastisch sein, wobei allerdings gesagt werden muß, daß DYNAMO sich für stochastische Simulationen grundsätzlich nicht eignet, da es nicht möglich ist, über mehrfach wiederholte Runs Statistiken zu berechnen. Jeder Run liefert einen Output und die Resultate sind dann im nächsten Run nicht mehr abrufbar. DYNAMO erlaubt nun (9-2) und (9-3) einfach zu spezifizieren, allerdings ist es nicht möglich, Befehle in DO-Schleifen zu geben. Jede Funktion von (9-2) und (9-3) muß einzeln angeführt werden. Weiters schränkt DYNAMO den Bereich zulässiger Funktionen für (9-2) und (9-3) stark ein - es können diese aber durch Einführen von Hilfsfunktionen beliebig verkoppelt werden. In Gleichung (9-3) sind für $m(.)$ nur die vier Grundrechnungsarten zulässig. Für (9-2) sind zusätzlich noch erlaubt Exponentialfunktion, Logarithmus, Quadratwurzel, Sinus, Cosinus, Maximum und Minimum. Außerdem können Funktionen der Art "wenn $A > B$ dann $y_{n-1} = \dots$ sonst $y_{n-1} = \dots$ " spezifiziert werden und schließlich kann $k(.)$ eine beliebige Funktion der Zeit sein, indem diese als Tabelle im Input spezifiziert wird.

Wohl ist das Anschreiben der Gleichungen (9-2) und (9-3) mitunter mühsamer in DYNAMO als in FORTRAN, der große Vorteil von DYNAMO besteht jedoch darin, daß keinerlei zusätzliche Befehle zum Berechnen von (9-2) und (9-3) in DYNAMO nötig sind. Auch der Printout ist sehr einfach zu spezifizieren, muß man doch nur die gewünschten Variablennamen angeben und kann diese dann in Form von Tabellen oder von graphischen Funktionen ausdrucken lassen.

Der Nachteil von DYNAMO besteht darin, daß man nicht stochastisch simulieren kann, daß das Anschreiben komplizierter zusammengesetzter Funktionen via Hilfsfunktion sehr aufwendig sein kann, daß für große s und p gemäß (9-4) das Fehlen von DO-Befehlen sich sehr störend bemerkbar macht und schließlich, daß die Integrationsmethode gemäß (9-3) numerisch doch zu simpel und

problematisch ist.

Von üblichen Differentialgleichungssystem 1.Ordnung der Form

$$\frac{dx}{dt} = m(x,t) \quad (9-6)$$

unterscheidet (9-5) bzw. (9-2) und (9-3) sich nur dadurch, daß die Funktion $y(x,t)$ gemäß (9-2) nicht explizit gegeben sein muß, sondern rekursiv berechnet wird, wodurch, wie gesagt, Funktionen zugelassen werden, welche bei expliziter Angabe, wie in (9-6), nicht angeführt werden können.

Auf die genaue Schreibweise von DYNAMO-Statements wollen wir erst im nächsten Kapitel im Zusammenhang mit einem Beispiel eingehen. Es soll einstweilen genügen, daß man Probleme der Form (9-2) und (9-3) mit DYNAMO lösen kann und solch ein Problem aus dem ökonomischen Bereich soll in Kapitel 10 diskutiert werden.

9.4. Literatur

A.L.Pugh, Dynamo II User's Manual, MIT PRESS, 1973

J.W.Forrester, Principle of Systems, Wright-Allen Press Inc.,
1971

10. EIN MAKROÖKONOMISCHES MODELL ZUR SIMULATION VON KONJUNKTURZYKLEN

10.1. Einleitung

In diesem Kapitel soll nun ein im Vergleich zu den beiden vorangehenden Modellen ungleich komplexeres System studiert werden, nämlich das makroökonomische Phänomen der Konjunkturzyklen. Auf Grund der zentralen Bedeutung, welche wirtschaftliche Veränderungen für den einzelnen haben, ist die Literatur darüber dementsprechend umfangreich. Existieren verbale Erklärungen in großer Vielfalt (siehe z.B. C.A.Danten & L.M. Valentine), so wird die Literatur, welche Konjunkturzyklen in mathematischer Sprache - durch Differenzen- oder Differentialgleichungen - darstellen, schon spärlicher. Um das mathematische Modell analysierbar zu machen, mußten jedoch häufig problematische Vereinfachungen der Realität vorgenommen werden. Einen Versuch, diese mathematischen Beschränkungen mittels eines Simulationsmodells abzubauen, stellt die Analyse von Produktionszyklen der Landwirtschaft in der Studie von D.L.Meadows dar. Im Folgenden soll ein ähnlicher Weg wie in dem Buch von D.L.Meadows gegangen werden, indem zunächst ein mathematisch formuliertes Modell zu einem Simulationsmodell "erweitert" wird. Daran anschließend, in Kapitel 10.3. wird das Modell in DYNAMO programmiert. In Kapitel 10.4. wird das dynamische Verhalten des Modells analysiert. Wie sich zeigen wird, ist im Vergleich zu den vorangehenden Modellen der programmiertechnische Teil bescheiden, hingegen die dahinterliegende Theorie und die Annahmen recht aufwendig.

10.2. Das ökonomische Modell

Ökonomische Theorien sind, zum Unterschied zu Theorien im naturwissenschaftlichen Bereich, weit davon entfernt von Ökonomen allgemein akzeptiert zu werden. Zu komplex ist die

Wirklichkeit als daß nicht für die Theorienbildung starke Vereinfachungen vorgenommen werden müssen. Eindeutige Experimente mit eindeutigen Resultaten zur Überprüfung von Theorien sind im ökonomischen Bereich naturgemäß fast unmöglich, weswegen die Frage, welcher Theorie nun der Vorzug zu geben ist, kaum entschieden werden kann. Auch Konjunkturzyklentheorien gibt es mehrere, teilweise ähnlicher, teilweise aber auch einander widersprechender Natur. So kommt es denn unweigerlich zur Klassifizierung von Theorien in z.B. monetaristische, keynesianische und marxistische um die, vielleicht bekanntesten zu nennen. Hier nun soll ein keynesianisches Modell erörtert werden, welches im Buch von J.T.Schwartz dargestellt ist. Das erörterte Modell beschreibt, wenn überhaupt, natürlich nur einen Ausschnitt der Realität.

Zwei ökonomische Merkmale charakterisieren das Modell:

- Konjunkturzyklen werden gemäß J.Keynes durch Überproduktion bzw., was dasselbe ist, durch Unterkonsumption erklärt.
- Daraus folgend spielt Geld und damit Preise als erklärender Faktor keine Rolle, taucht daher im Modell gar nicht auf, d.h. Güter werden mengenmäßig wohl in Geldeinheiten bei gegebenen Preisen beschrieben, aber diese Preise sind exogen vorgegeben und für das Modell konstant.

In einem weiteren Buch von J.T.Schwartz, Theory of Money, wird auch ein Modell erstellt, welches Preisveränderungen im Modell erklärt. Der Einfachheit halber soll hier das Basismodell dargestellt werden. Die beiden Schwartz'schen Modelle sind einander aber soweit ähnlich, daß die Erweiterung des hier darzustellenden Simulationsmodells auf das Modell mit variablen Preisen ohne große Probleme möglich ist.

Wir betrachten die gesamte wirtschaftliche Produktion aggregiert in zwei Sektoren, die Investitionsgüterindustrie und die Konsumgüterindustrie. Es sei gleich erwähnt, daß diese Simplifizierung

nicht notwendig ist und bei Schwartz auch nicht gemacht wird, er läßt beliebige aber endlich viele Sektoren zu. Die Vereinfachung wird ausschließlich vorgenommen, um den programmier-technischen Aufwand vertretbar zu gestalten. Unter Investitionsgüter seien hier Produkte verstanden, welche erzeugt werden, um die Produktion von, wiederum, Investitionsgütern und auch Konsumgütern zu ermöglichen, also Maschinen, welche nicht konsumiert sondern primär der Produktion von Konsumgütern dienen. Konsumgüter hingegen sind Güter, welche dem Letztverbraucher, also dem Menschen unmittelbar zugute kommen. Natürlich ist nicht bei jedem Gut klar, ob es ein Konsum- oder Investitionsgut darstellt, eine Unschärfe, die bei mehrfacher Verwendung eines Gutes unvermeidlich ist (z.B. bei Energie). Außerdem suggeriert die Aufspaltung in die oben genannten Sektoren, daß Investitions- bzw. Konsumgüterindustrien getrennte Einheiten sein müssen, was in der Realität zweifelsohne nicht zutrifft, kurzum unser Klassifikationsschema weist deutliche Schwächen auf, welche aber für das Modell nicht von prinzipieller Natur sind.

Jedem Sektor wird nun auch ein Lager zugeteilt, dessen Aufgabe es ist, als Puffer zwischen Produktion und Verkauf zu dienen.

Als dritten "Sektor" benötigen wir nun noch Arbeitskräfte, welche die Produktion in den ersten beiden Sektoren erst ermöglichen, welche aber auch als Käufer für Konsumgüter auftreten. Diese drei Produktionssektoren sollen nun im Folgenden miteinander in Beziehung gesetzt werden.

In jedem der drei Sektoren führen wir nun eine Einheit zum Quantifizieren der Güter bzw. der Arbeit ein.

Sei

- C_0 Arbeitseinheit (z.B. eine Arbeitsstunde)
- C_1 Einheit Investitionsgüter
- C_2 Einheit Konsumgüter .

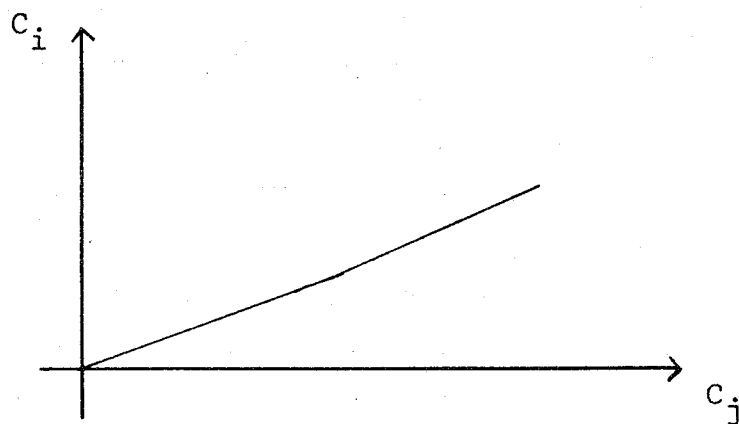
Diese Einheiten können sowohl warenmäßig, d.h. in Stück ausgedrückt werden, oder, was in unserem Fall notwendig ist, in Geld (z.B. 1 Mill. ÖS) ausgedrückt werden. Letzteres ist aber nur bei konstanten Preisen möglich. Um jedoch eine Vielzahl verschiedener Güter unter eines zu subsummieren, bedarf es einer einheitlichen Größe, eben Geld. In welcher Form die Einheiten tatsächlich spezifiziert werden, muß uns jetzt aber nicht interessieren. Wir definieren nun die Größen π_{ij} ($i, j=0,1,2$) folgendermaßen:

Sei π_{ij} die Menge an C_j -Einheiten, die verbraucht wird, um eine Einheit von C_i zu produzieren.

Die π_{ij} setzen also die Inputs für die Produktion mit deren Outputs in Beziehung.

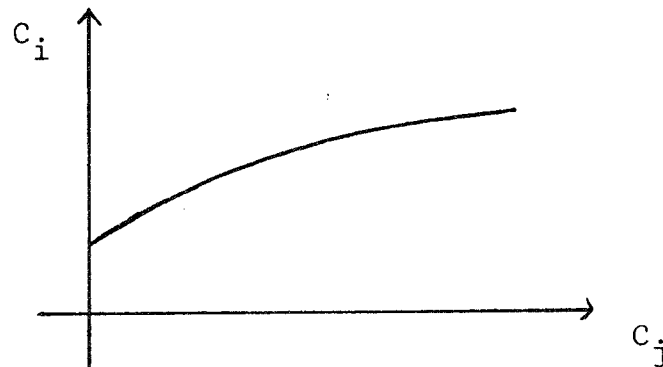
Einige Probleme der Input-Output Methode sollen hier nicht unerwähnt bleiben.

- Die π_{ij} sind unabhängig von der Produktionshöhe angenommen, d.h. π_{ij} ist unabhängig davon ob eine oder eine Million Einheiten von C_i produziert werden. Die Beziehungen zwischen Input und Output ist daher linear, d.h. es wird der in Figur 10-1 dargestellte Zusammenhang zwischen C_j und C_i angenommen.



Figur 10-1

Tatsächlich ist der Zusammenhang meistens von der Art, wie in Figur 10-2 dargestellt.



Figur 10-2

Zur Rechtfertigung des linearen Modells ist zu sagen, daß in einem genügend kleinen Bereich um einen vorgegebenen Produktionslevel, die lineare Beziehung die in Figur 10-2 dargestellte nicht-lineare Relation ausreichend genau approximiert.

Grundsätzlich existieren bereits nichtlineare Input-Output Modelle und auch in unserem Simulationsmodell kann statt der linearen eine nichtlineare Relation hergestellt werden.

- Da die π_{ij} unabhängig von der Produktionshöhe sind, wird auch nicht berücksichtigt, daß bestimmte Inputs beschränkt und nicht reproduzierbar sind (insbesondere Boden und Rohstoffe). Daher eignet sich das Modell nicht, falls Fragestellungen im Zusammenhang mit z.B. Rohstoffverbrauch untersucht werden sollen.
- Schließlich impliziert das Modell, daß jede Input- bzw. jede Outputkombination möglich ist, d.h. Produktionsprozesse, welche mehrere Güter in einem bestimmten Verhältnis zueinander erzeugen, werden in diesem Modell ignoriert (z.B. Koks- Gas oder, schlimmer,

Farbstoffe- Abwässer). Dieses Problem kommt allerdings erst zum Tragen, falls viele Produktionssektoren betrachtet werden.

- In unserem Modell werden wir annehmen, daß π_{ij} auch von der Zeit abhängt.

Nicht beschäftigen werden wir uns mit der Frage, wie π_{ij} gemessen bzw. geschätzt werden kann. Im vorliegenden 3-Sektoren Modell sollen folgende zusätzliche, aber nicht notwendige Annahmen über π_{ij} getroffen werden:

- Um Arbeit zu produzieren, bedarf es nur Konsumgüter, daraus folgt

$$\pi_{00} = 0 \quad , \quad \pi_{01} = 0$$

- Um Konsumgüter zu produzieren, bedarf es nur Investitionsgüter und Arbeit

$$\pi_{22} = 0$$

- Um Investitionsgüter zu produzieren, bedarf es nur Investitionsgüter und Arbeit

$$\pi_{12} = 0 \quad .$$

π_{02} stellt also die Menge an Konsumguteinheiten zur Produktion von Arbeit dar. π_{02} hat die Dimension Konsumguteinheit pro Arbeitseinheit und stellt nichts anderes dar, als den Anteil der Löhne, welcher nicht gespart wird, sondern zum Konsum verwendet wird. (gemäß J.Keynes ist dieser direkt proportional zum Lohn).

Sei nun die Produktion pro gewählter Zeiteinheit (z.B. Monat) a_0 (für Arbeit), a_1 (für Investitionsgüter) und a_2 . Am Ende der Produktionsperiode erhält man (Produktionshöhe minus während der Produktion verbauchte Güter):

$$a_0 = \sum_{i=0}^2 a_i \pi_{i0} \quad (10-1)$$

$$a_1 = \sum_{i=0}^2 a_i \pi_{i1} \quad (10-2)$$

$$a_2 = \sum_{i=0}^2 a_i \pi_{i2} \quad (10-3)$$

Da Arbeit weder gelagert noch mehr verbraucht werden kann, als zur Verfügung steht, muß für (10-1) gelten

$$a_0 = \sum_{i=0}^2 a_i \pi_{i0} = 0 \quad (10-4)$$

Setzt man (10-4) in (10-2) und (10-3) ein, so erhält man

$$\begin{aligned} a_1 &= (a_1 \pi_{11} + a_2 \pi_{21}) \\ a_2 &= (a_1 \pi_{10} + a_2 \pi_{20}) \pi_{02} \end{aligned} \quad (10-5)$$

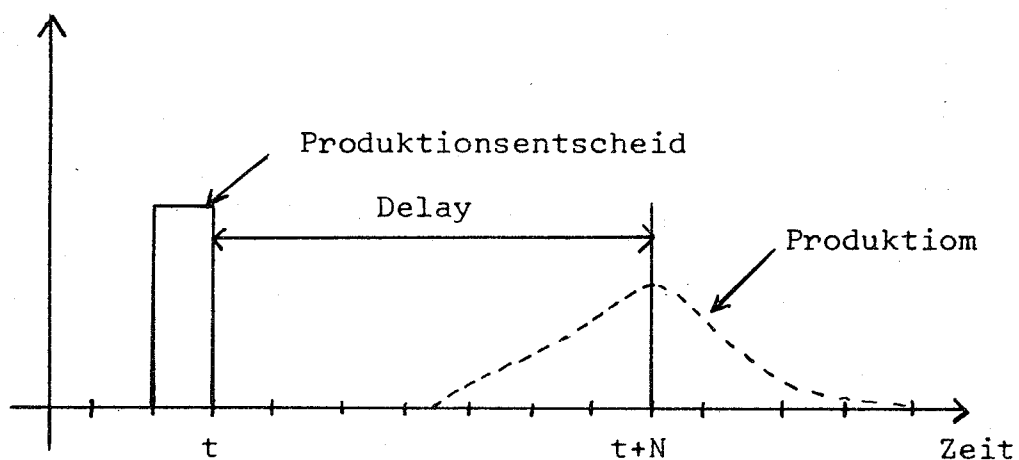
Ist der Ausdruck (10-5) positiv, so wird mehr produziert als verbraucht, d.h. die Wirtschaft ist expandierend, andernfalls schrumpfend. Interessant ist natürlich der erste Fall und für reale Werte von π_{ij} , sollte (10-5) auch stets positiv sein. Der jeweils zweite Term in (10-5) ist der Teil, der verkauft wurde.

Betrachten wir nun den Lagerbestand jedes Sektors etwas näher: Der Lagerbestand des Sektors i zum Zeitpunkt t , $b_i(t)$, ergibt sich aus dem alten Lagerbestand plus der Überschußproduktion (10-5) minus dem Konsum $e(t-1)$ bei den Konsumgütern. Dieser Konsum $e(t)$ ist eine exogen definierte Größe und beschreibt den Konsum des Staats und der Unternehmer. Dann gilt:

$$\begin{aligned} b_1(t) &= b_1(t-1) + a_1(t-1) [1 - \pi_{11}] - a_2(t-1) \pi_{21} \\ b_2(t) &= b_2(t-1) - a_1(t-1) \pi_{10} \pi_{02} + a_2(t-1) [1 - \pi_{20} \pi_{02}] - e(t-1) \end{aligned} \quad (10-6)$$

Es gilt nun, das Verhalten der beiden Sektoren bezüglich den Entscheidungen über das Produktionsniveau festzulegen. Es wird angenommen, daß das Produktionsniveau im wesentlichen vom Lagerbestand abhängt. Ausgehend davon, daß jeder Unternehmer seinen Lagerbestand so hoch halten möchte, daß er stets lieferbereit ist, andererseits so niedrig wie möglich halten möchte, um keine Gewinnverluste durch brachliegendes Kapital in Kauf zu nehmen, scheint es sinnvoll eine optimale Lagergröße als Entscheidungsparameter anzunehmen. Ist mehr als erwünscht am Lager, wird die Produktion gedrosselt, ist es weniger, wird die Produktion ausgeweitet. Die Entscheidung über die Produktionshöhe wird allerdings erst nach einiger Zeit die Lagergröße beeinflussen, dauert es doch einige Zeit bis das Produkt den Produktionsprozeß verläßt. Für diese Zeitspanne oder Produktionsdelay muß der Unternehmer daher eine Schätzung über den Verkauf haben. Umgekehrt kann er auch prognostizieren, was er während dieses Produktionsdelays auf Grund früherer Produktionsentscheide produzieren wird. Diese beiden Schätzungen zusammen mit einer optimalen Lagergröße ergibt die Produktionsentscheidung.

Betrachten wir also einen Produktionsdelay von N Zeiteinheiten, d.h. Produktionsentscheide zum Zeitpunkt t werden um den Zeitpunkt $t+N$ realisiert (siehe Figur 10-3).



Figur 10-3

Dann benötigt man für die Zeitpunkte $t+1, t+2, \dots, t+(N-1)$ Prognosen über den Produktionsoutput. Diese Prognose, so nehmen wir an, sei die Summe über die Produktionsentscheide in den Zeitpunkten $t-1, t-2, \dots, t-(N-1)$. Sei also $d_i(t)$ der Produktionsentscheid in Sektor i zum Zeitpunkt t , so lautet die erwartete Produktion $ERP_i(t)$ in Sektor i zum Zeitpunkt t für die nächsten N_i-1 Zeitperioden

$$ERP_i(t) = \sum_{k=1}^{N_i-1} d_i(t-k) \quad (10-7)$$

Benötigt wird auch eine Schätzung über den Verkauf in den nächsten N_i Zeitperioden. Ausgehend vom Verkauf der letzten N_i Perioden im Sektor i , nämlich

$$V_1(t) = \sum_{k=1}^{N_1} [a_1(t-k)\pi_{11} + a_2(t-k)\pi_{21}]$$

$$V_2(t) = \sum_{k=1}^{N_1} [a_1(t-k)\pi_{10}\pi_{02} + a_2(t-k)\pi_{10}\pi_{02} + e(t-k)] \quad (10-8)$$

Es soll angenommen werden, daß der erwartete Verkauf eine lineare Extrapolation der vergangenen Verkäufe darstellt. Es gelte für den erwarteten Verkauf $EERV_i(t)$

$$EERV_i(t) = V_i(t) + \frac{ERV_i(t-1)}{2} \quad (10-9)$$

$$ERV_i(t) = V_i(t) - V_i(t-1) + \frac{ERV_i(t-1)}{2} \quad (10-9a)$$

Setzt man (10-9a) in (10-9) ein, so ergibt sich mit $ERV_i(0)=0$

$$EERV_i(t) = V_i(t) + \sum_{j=1}^t \frac{V_i(j) - V_i(j-1)}{2^{t-j+1}}$$

Der erwartete Verkauf wird daher als der Verkauf der Vorperiode korrigiert um einen Trend angenommen. Ist insbesondere der Trend konstant, d.h. gilt

$$V_i(j) - V_i(j-1) = C_i$$

für alle $j \geq 1$, so wird für $t \rightarrow \infty$ der Trend richtig geschätzt, nämlich durch

$$\lim_{t \rightarrow \infty} \sum_{j=1}^t \frac{C_i}{2^{t-j+1}} = \lim_{t \rightarrow \infty} \sum_{j=1}^t \frac{C_i}{2^j} = C_i$$

Diesem Modell für den erwarteten Verkauf liegt die Annahme zugrunde, daß Wirtschaftseinheiten sehr einfache Prognosen vornehmen, z.B. unter Annahme konstanter Wachstums- (oder Schrumpfungsraten). In (10-9), (10-9a) werden daher vergangene Trends, gewichtet auf Grund ihrer Aktualität, zur linearen Extrapolation verwendet.

Nun gilt es noch, einen Ausdruck für den erwünschten optimalen Lagerbestand $OPL_i(t)$ des Sektors i für den Zeitpunkt $t+N$ zum Entscheidungszeitpunkt t zu finden. Er laute

$$OPL_i(t) = H_i + C_i \cdot ERV_i(t), \quad H_i, C_i \geq 0 \quad (10-10)$$

Der erwünschte Lagerbestand soll also aus einem festen Grundstock sowie einem vom erwarteten Verkauf abhängigen variablen Teil bestehen.

Somit ergibt sich die Produktionsentscheidung

$$d_i(t) = ERV_i(t) + OPL_i(t) - ERP_i(t) - b_i(t) \quad (10-11)$$

oder in Worten

Produktion = Erwarteter Verkauf + Erwünschter Lagerbestand -
 - Momentaner Lagerbestand - Erwartete Produktion .

Ausständig ist nun noch die tatsächliche Produktion $a_i(t)$,
 welche sich gemäß Figur 10-3 aus den vorangehenden Produktions-
 entscheiden mit einem Delay von N Zeitperioden ergibt. Die
 Formel hierfür werden wir erst im nächsten Kapitel besprechen.

Für den exogenen Konsum $e(t)$ sollen zwei Varianten simuliert
 werden. In einem Fall sei $e(t)$ konstant

$$e(t) = e(t-1)$$

im anderen Fall soll studiert werden, ob in unserem Modell
 antizyklisches Verhalten der Regierung bei Budgetausgaben
 stabilisierend wirken kann. Hierzu wird folgende Politik
 angenommen: Falls die Produktion wächst, soll $e(t)$ ebenfalls
 wachsen, andernfalls soll $e(t)$ konstant bleiben, d.h.

$$\left. \begin{array}{l} a_1(t) > a_1(t-1) \\ \text{oder } a_2(t) > a_2(t-1) \end{array} \right\} \Rightarrow \begin{array}{l} e(t) = w \cdot e(t-1) \\ w > 1 \end{array} \quad (10-12)$$

$$\text{sonst} \quad e(t) = e(t-1)$$

Untersuchen wollen wir schließlich noch den Fall zeit-
 abhängiger Input-Outputkoeffizienten. Es soll angenommen
 werden

$$\pi_{ij}(t) = p_{ij} \cdot \pi_{ij}(t-1)$$

wobei die Löhne steigen sollen, d.h.

$$p_{0j} > 1 \quad j=1,2 \quad (10-13)$$

und die Produktivität auch, d.h.

$$p_{ij} < 1 \quad i,j=1,2$$

Wir haben nun ein geschlossenes System erstellt, in dem wir nur mehr die Anfangswerte definieren müssen. Wie realistisch das Modell nun ist, sollen einerseits die Resultate zeigen, ist aber andererseits auch eine Anschauungsfrage, nämlich inwieweit das beschriebene Unternehmerverhalten, vor allem Beziehungen (10-7)-(10-11) tatsächliches Unternehmerverhalten wiedergibt. Zum Unterschied zum mathematischen Modell von J.T.Schwartz, können bei der Simulation allerdings weitgehend beliebige Verhaltensgleichungen eingeführt werden. Auch die von uns beschriebenen weichen in einigen Punkten von den Schwartz'schen Annahmen ab. Im nächsten Kapitel soll nun das DYNAMO-Programm spezifiziert werden.

10.3. Das DYNAMO-Programm

Nach einigen erläuternden Vorbemerkungen in Kapitel 9.3. über DYNAMO, soll nun ein Programm in dieser Sprache erstellt werden. Genausowenig wie bei SIMSCRIPT kann hier eine komplette Beschreibung der Sprache erfolgen, hiezu muß auf das Manual von A.L.Pugh verwiesen werden. Vielmehr soll das Prinzip von DYNAMO am Umsetzen der in Kapitel 10.2. aufgeführten Differenzengleichungen dargestellt werden. DYNAMO erlaubt es, die in Kapitel 10.2. definierten Gleichungen in sehr ähnlicher Form darzustellen. In jeder Zeile (Lochkarte) wird eine Differenzengleichung definiert. Zur Spezifizierung muß zunächst der Gleichungstyp durch eine Nummer beschrieben werden, wobei jeder Nummer in eindeutiger Weise ein Gleichungstyp entspricht. Weiters muß die so definierte Größe näher erläutert werden, durch den Code

L	Level
R	Rate
A	Hilfsgröße
C	Konstante
N	Initialisierung (Anfangswert)

Daran anschließend erfolgt ab Spalte 7 der Lochkarte die eigentliche Gleichung.

Soll, als Beispiel, die Gleichung

$$V(t) = V(t-1) + \Delta t \cdot (a(t-1) + b(t-1))$$

dargestellt werden, so lautet die DYNAMO-Schreibweise

$$\begin{array}{ccc} 1L & & V.K=V.J+(DT)(A.J+B.J) \\ \uparrow & & \uparrow \\ \text{Spalte 1} & & \text{Spalte 7} \end{array}$$

Wie bereits in Kapitel 9.3. erwähnt, steht nur eine begrenzte Anzahl Gleichungstypen zur Verfügung, die aber über Hilfsgrößen (Typ A) beliebig verknüpft werden können. Die einzelnen Gleichungsnummern wollen wir nicht auflisten, sie sind dem Anhang des Buches von A.L.Pugh zu entnehmen.

Beginnen wir nun mit der Transformierung der Gleichungen aus Kapitel 10.2.

Verkauf:

Aus (10-5) bzw. (10-6) läßt sich der Verkauf ableiten, nämlich

$$\begin{array}{ll} a_1(t)\pi_{11} + a_2(t)\pi_{21} & \text{in Sektor 1} \\ (a_1(t)\pi_{10} + a_2(t)\pi_{20})\pi_{02} + e(t) & \text{in Sektor 2.} \end{array}$$

Die entsprechenden DYNAMO-Gleichungen lauten

$$\begin{array}{l} 15A \quad VK1.K=(A1.JK)(P11.JK)+(A2.JK)(P21.JK) \\ 17A \quad VK2.K=(A1.JK)(P10.JK)(P02.JK)+(A2.JK)(P20.JK)(P02.JK)+ \\ \quad \quad \quad +(E.JK)(1)(1) \end{array}$$

Größen mit Doppelindizes JK bedeuten Rates.

Lagerbestand:

Aus (10-6) folgt für den Lagerbestand

$$6R \quad B1.KL = BB1.K$$

$$8A \quad BB1.K = B1.JK + A1.JK - VK1.K$$

$$6R \quad B2.KL = BB2.K$$

$$8A \quad BB2.K = B2.JK + A2.JK - VK2.K$$

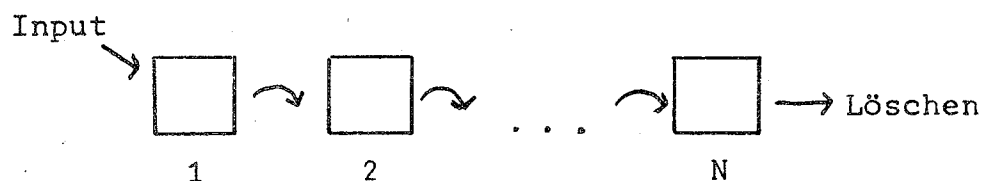
$$6N \quad B1 =$$

$$6N \quad B2 =$$

Die letzten beiden Gleichungen definieren die Anfangswerte der Lagerbestände. Der Umweg zur Bestimmung von B1 bzw. B2 über die Hilfsvariablen BB1 und BB2 wurde gewählt, da die Werte von B1.KL bzw. B2.KL noch in anderen Gleichungen benötigt werden, wo allerdings auf der rechten Seite der Gleichung nur die Indizes JK, also die Werte der Vorperiode stehen dürfen, wohl aber der Index K erlaubt ist.

Erwartete Produktion in den nächsten N-1 Perioden:

In (10-7) ist die Produktionsprognose aufgestellt. Um dies in DYNMA0 zu übersetzen benötigt man den Begriff "boxtrain". Es handelt sich hierbei um eine indizierte Größe in die unter Index 1 der aktuellste Wert und beim höchsten Index der älteste Wert einer Variablen steht. Wird ein neuer Wert unter Index 1 hineingeschrieben, so werden alle alten Werten um einen Index nach oben geschoben und der letzte Wert gelöscht (siehe auch Figur 10-4)



Figur 10-4

```

37B  BX1=BOXLIN(N1,1)
37B  BX2=BOXLIN(N2,1)
36N  BX1=BOXLOAD(DT,A1)
36N  BX2=BOXLOAD(DT,A2)
53A  ERP1.K=SUM1(N1,BX1.K)
53A  ERP2.K=SUM1(N2,BX2.K)
3L   BX1*1.K=BX1*1.J+(DT)*(1/DT)*(D1.J-BX1*1.J)
3L   BX2*1.K=BX2*1.J+(DT)*(1/DT)*(D2.J-BX2*1.J)

```

Gleichungen 37B geben an, daß es sich um boxtrains mit N1 bzw. N2 Elementen handelt und daß in jedem Zeitschritt die Werte um einen Index verschoben werden. In 36N wird initialisiert gemäß

$$d_i(-k) = d_i(0) \cdot DT \quad k=1, \dots, N_i$$

wobei DT die gewählte Schrittlänge darstellt ($DT=t_N-t_{N-1}$).

Durch 53A wird die Beziehung (10-7) berechnet und bei 3L die aktuellen Werte von $d_i(t)$ eingefügt.

Erwarteter Verkauf in den nächsten N-1 Perioden:

Die DYNAMO-Statements sind ganz analog den vorangehenden

```

37B  PX1=BOXLIN(.,1)
37B  PX2=BOXLIN(.,1)
36N  PX1=BOXLOAD(DT,.)
36N  PX2=BOXLOAD(DT,.)
3L   PX2*1.K=PX2*1.J+(DT)*(1/DT)*(VK2.J-PX2*1.J)
3L   PX1*1.K=PX1*1.J+(DT)*(1/DT)*(VK1.J-PX1*1.J)
53A  V1.K=SUM1(.,PX1.K)
53A  V2.K=SUM1(.,PX2.K)
6R   VV1.KL=V1.K
6R   VV2.KL=V2.K
27A  EERV1.K=(ERV1.JK/2)+V1.K
27A  EERV2.K=(ERV2.JK/2)+V2.K
7R   ERV1.KL=EERV1.K-VV1.JK
7R   ERV2.KL=EERV2.K-VV2.JK
6N   ERV1=0
6N   ERV2=0

```

Die letzten Statements stellen die Berechnung von (10-9) dar, ergänzt um die Anfangsbedingungen.

Optimaler Lagerbestand

Aus (10-10) folgt

14A $OPL1.K = H1 + (C1)(EERV1.K)$

14A $OPL2.K = H2 + (C2)(EERV2.K)$

C $H1 = /C1 = /H2 = /C2 =$

Produktionsentscheidung

Gemäß (10-11) ergibt sich diese mit

9A $DD1.K = EERV1.K + OPL1.K - ERP1.K - BB1.K$

9A $DD2.K = EERV2.K + OPL2.K - ERP2.K - BB2.K$

56A $D1.K = \text{MAX}(0, DD1.K)$

56A $D2.K = \text{MAX}(0, DD2.K)$

Durch 56A wird eine negative Produktion verhindert.

Produktionsoutput:

DYNAMO stellt eine spezielle Delay-Funktion gemäß Figur (10-3) zur Verfügung, nämlich

39R $A1.KL = \text{DELAY3}(D1.K, DEL1)$

39R $A2.KL = \text{DELAY3}(D2.K, DEL2)$

12N $A1 = (DT).(.)$

12N $A2 = (DT).(.)$

C $DEL1 = /DEL2 =$

DEL1 bzw. DEL 2 gibt die Delaylänge an, wobei aber gelten muß

Delaylänge ≥ 3 .

d.h. Gleichung 39R bewirkt einen Mindestdelay von 3 Zeiteinheiten, weswegen man die Zeiteinheiten bei gegebenen Delay dementsprechend wählen muß. Es muß natürlich gelten

$$\text{DEL1-1} = \text{N1}$$

$$\text{DEL2-1} = \text{N2} .$$

Wir werden als Delay etwa 2-4 Monate annehmen, woraus folgt, daß eine sinnvolle Zeiteinheitwahl einen halben Monat darstellt. Daher bedeute

$$\text{DT}=1$$

in DYNAMO ein halbes Monat und es gelte bei der Wahl von DT stets

$$0 < \text{DT} \leq 1.$$

(10-14)

DT=1 bedeutet aber auch, daß alle halbe Monat der Lagerbestand geprüft und ein Produktionsentscheid gefällt wird. Da a_i die Produktion/Periode bedeutet, ist sie natürlich von (DT) abhängig und wird dementsprechend initialisiert.

Spezifikation der Input-Output Koeffizienten und des exogenen Konsums: Zunächst sollen alle diese Größen unabhängig von der Zeit sein, woraus folgt

$$6R \quad E.KL=E.JK$$

$$12N \quad E=(DT).(.)$$

$$6R \quad P11.KL=P11.JK$$

$$6R \quad P21.KL=P21.JK$$

$$6R \quad P10.KL=P10.JK$$

$$6R \quad P20.KL=P20.JK$$

$$6R \quad P02.KL=P02.JK$$

$$6N \quad P11=$$

$$6N \quad P21=$$

$$6N \quad P10=$$

$$6N \quad P20=$$

$$6N \quad P02=$$

Damit wäre nun das DYNAMO-Programm vollständig spezifiziert.

Lohnhöhe

15A $AO.K = (A1.JK)(P10.JK) + (A2.JK)(P20.JK)$

Diese Gleichung ergibt sich dabei aus Beziehung (10-4). Bis auf die Outputbefehle ist damit das DYNAMO-Programm fertig:
Das komplette Programm-listing lautet:

RUN OEKMO1

NOTE

NOTE VERKAUF

NOTE

15A $VK1.K = (A1.JK)(P11.JK) + (A2.JK)(P21.JK)$

17A $VK2.K = (A1.JK)(P10.JK)(P02.JK) + (A2.JK)(P20.JK)(P02.JK) + (E.JK)(1)(1)$

NOTE

NOTE LAGERBESTAND

NOTE

6R $B1.KL = BB1.K$

8A $BB1.K = B1.JK + A1.JK - VK1.K$

6R $B2.KL = BB2.K$

8A $BB2.K = B2.JK + A2.JK - VK2.K$

6N $B1 = 6.6$

6N $B2 = 19.2$

NOTE

NOTE ERWARTETE PRODUKTION

NOTE

37B $BX1 = BOXLIN(8, 1)$

37B $BX2 = BOXLIN(6, 1)$

36N $BX1 = BOXLOAD(DT, 6)$

36N $BX2 = BOXLOAD(DT, 17.5)$

53A $ERP1.K = SUM1(8, BX1.K)$

53A $ERP2.K = SUM1(6, BX2.K)$

3L $BX1*1.K = BX1*1.J + (DT)(1/DT)(D1.J - BX1*1.J)$

3L $BX2*1.K = BX2*1.J + (DT)(1/DT)(D2.J - BX2*1.J)$

NOTE

NOTE ERWARTETER VERKAUF

NOTE

37B $PX1 = BOXLIN(9, 1)$

37B $PX2 = BOXLIN(7, 1)$

36N $PX1 = BOXLOAD(DT, 6)$

36N $PX2 = BOXLOAD(DT, 17.5)$

3L $PX2*1.K = PX2*1.J + (DT)(1/DT)(VK2.J - PX2*1.J)$

3L $PX1*1.K = PX1*1.J + (DT)(1/DT)(VK1.J - PX1*1.J)$

53A $V1.K = SUM1(9, PX1.K)$

53A $V2.K = SUM1(7, PX2.K)$

6R $VV1.KL = V1.K$

6R $VV2.KL = V2.K$

27A $EERV1.K = (ERV1.JK/2) + V1.K$

27A $EERV2.K = (ERV2.JK/2) + V2.K$

7R $ERV1.KL = EERV1.K - VV1.JK$

7R $ERV2.KL = EERV2.K - VV2.JK$

6N $ERV1 = 0$

6N $ERV2 = 0$

NOTE

NOTE ERWUENSCHTER LAGERBESTAND

NOTE

14A $OPL1.K = H1 + (C1)(EERV1.K)$

14A $OPL2.K = H2 + (C2)(EERV2.K)$

C $H1 = 3/H2 = 9/C1 = 0.4/C2 = 0.5$

NOTE
NOTE
NOTE

FESTLEGEN DER PRODUKTIONSHOEHE

9A DD1.K=EERV1.K+OPL1.K-ERP1.K-BB1.K
9A DD2.K=EERV2.K+OPL2.K-ERP2.K-BB2.K
56A D1.K=MAX(0,DD1.K)
56A D2.K=MAX(0,DD2.K)

NOTE
NOTE
NOTE

PRODUKTIONSOUTPUT

39R A1.KL=DELAY3(D1.K,DEL1)
39R A2.KL=DELAY3(D2.K,DEL2)
12N A1=(DT)(6)
12N A2=(DT)(17.5)
C DEL1=9/DEL2=7

NOTE
NOTE
NOTE

EXOGENER KONSUM (STAAT, UNTERNEHMER)

6R E.KL=E.JK
12N E=(DT)(6.7)

NOTE
NOTE
NOTE

DYNAMIK DER INPUT-OUTPUT KOEFFIZIENTEN

6R P11.KL=P11.JK
6R P21.KL=P21.JK
6R P10.KL=P10.JK
6R P20.KL=P20.JK
6R P02.KL=P02.JK
6N P11=0.33
6N P21=0.23
6N P10=0.5
6N P20=0.45
6N P02=1

NOTE
NOTE
NOTE

LOHNHOEHE

15A A0.K=(A1.JK)(P10.JK)+(A2.JK)(P20.JK)

NOTE
NOTE

PRINT- UND PLOTBEFEHLE

PRINT 1)B1,B2/2)A1,A2/3)VK1,VK2/4)ERP1,ERP2/5)V1,V2/6)ERV1,ERV2/7)D1,D2/
X1 8)EERV1,EERV2/9)OPL1,OPL2
PLOT A0=0,A1=*,A2=X
SPEC DT=1/LENGTH=299/PRTPER=0/PLTPER=3

10.4. Resultate und Kritik

Die Durchrechnung des Modells ist in Figur 10-4 dargestellt. Das Verhalten der drei Sektoren, Investitions-, Konsumgüter und Arbeit entspricht dem im Buch von J.T.Schwartz abgeleiteten, wiewohl einige Modellveränderungen vorgenommen wurden, in der Hoffnung, etwas realistischere Annahmen zu treffen. Und zwar wurde im Unterschied zu J.T.Schwartz der Delay zwischen Entscheidung und Produktion eingeführt sowie ein anderes Modell der Verkaufserwartung gewählt. Die Daten, das sei hier gesagt, sind indes keine realistischen Werte. Insofern eignet sich das Modell zweifelsohne nicht, um Prognosen über den Wirtschaftsverlauf zu tätigen. Es soll aber die Frage untersucht werden, ob eine gezielte Beeinflussung des exogenen Konsums im Modell den "Modellkonjunkturverlauf" beeinflussen kann, wobei von der Annahme ausgegangen wird, daß der exogene Konsum durch den Staat wesentlich steuerbar ist. Sei nun der Konsum gleich in DYNAMO-Schreibweise durch folgendes Differenzengleichungssystem definiert:

39R	$E.KL = DELAY3(AAO.JK.15)$	
6R	$AOH.KL = AO.K$	
40A	$WAO.K = (1) + (1/AOH.JK)(AO.K - AOH.JK)$	
56A	$WAOH.K = MAX(1, WAO.K)$	(10-14)
30R	$AAO.KL = (AAO.JK)SQRT(WAOH.K)$	
12N	$E = (DT)(6.7)$	
6N	$AAO = E$	

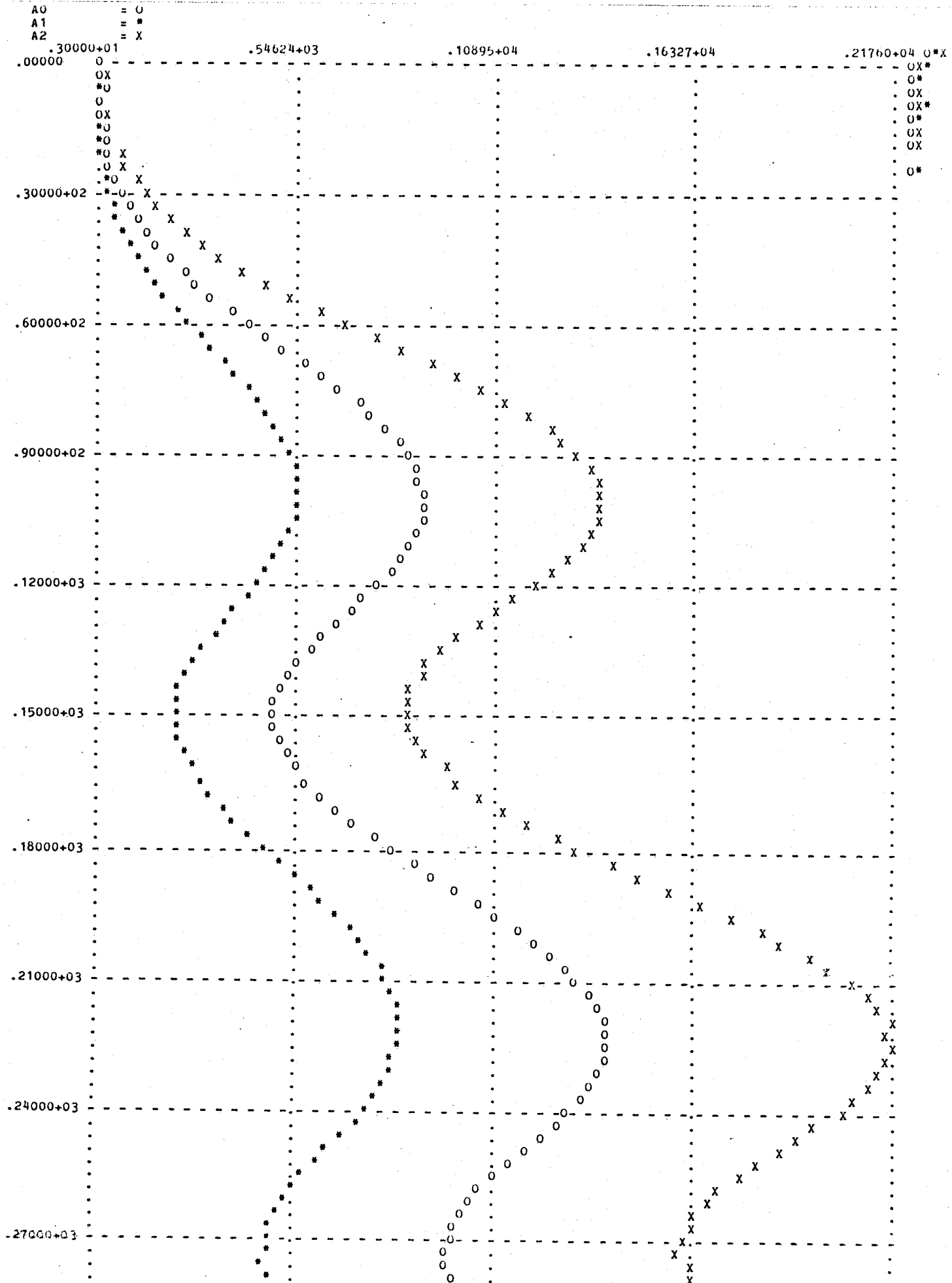
Das bedeutet, daß der Staat seine Ausgaben proportional zur Quadratwurzel der Löhne steigert, bzw., falls die Löhne oder besser, bei konstanten Löhnen, die Anzahl Arbeitender sinkt, daß in diesem Fall die Staatsausgaben konstant bleiben, also nicht sinken. Die Simulation dieses Modells ergibt einen Konjunkturverlauf gemäß Figur 10-5. Hierbei fällt auf, daß im Vergleich zu Figur 10-4 die Aufschwungsphase länger dauert, die Schrumpfung nicht so stark ist. Daß die Produktion in Figur 10-5 langfristig wächst, wohingegen sie in Figur 10-4 konstant bleibt, ist eine klare Folge des wachsenden exogenen Konsums.

Als letzte Variante soll noch der Fall, daß die Löhne mit der Zeit variieren untersucht werden. Hierbei wird für den Input-Output Koeffizienten π_{02} angenommen

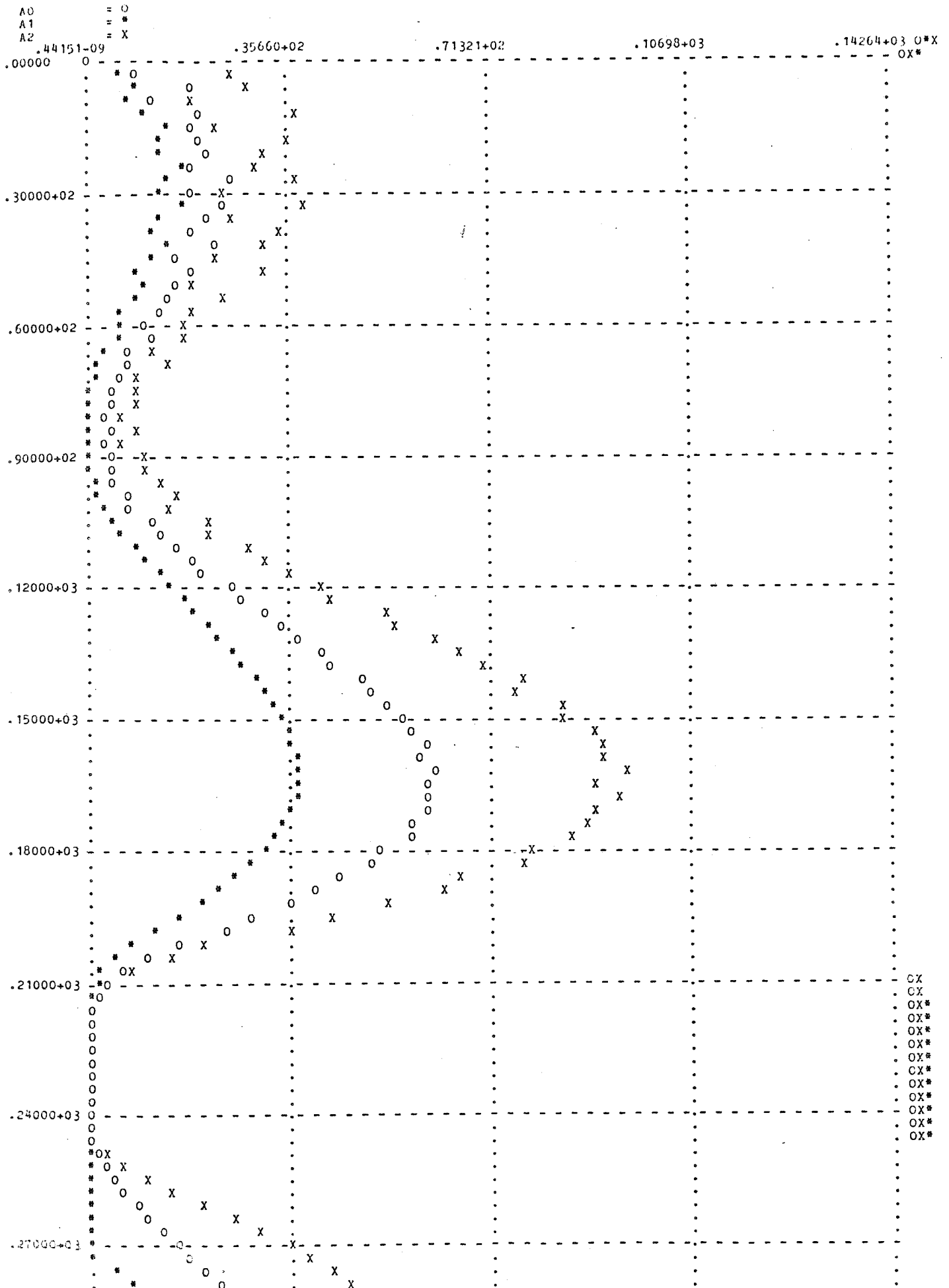
$$7R \quad P02.KL=P02.JK+0.002$$

(10-15)

d.h. die Löhne sollen nun einen konstanten Betrag pro Zeitperiode wachsen. Falls alle übrigen Gleichungen im ursprünglichen Modell unverändert bleiben ergibt sich als Resultat Figur 10-6. Wie in Figur 10-5 zeigt sich, daß die Wachstumsphase länger dauert als die Kontraktionsphase im Unterschied zu Figur 10-4. Allerdings sind die Rezessionen sehr ausgeprägt.



Figur 10-5



Figur 10-6

Da das ökonometrische Problem des Schätzens der Parameter den Rahmen einer Einführung in die Simulation sprengen würde, soll auf diesen Punkt auch nicht weiter eingegangen werden. Natürlich sind noch bei weitem nicht alle interessanten Varianten des Modells hier analysiert worden. So wäre sicher zu untersuchen, wie sich der Einfluß nichtlinearer Input-Output Beziehungen auswirkt.

Zum Unterschied zu den vorangehenden Kapitel über diskrete Simulation stecken hier die Hauptprobleme bei der Simulation eher im Inhaltlichen (bei der Formulierung des Modells) als im Formalen (beim Programmieren und Analysieren - insbesondere bei stochastischen Simulationen).

10.5. Literatur

A.L.Pugh, DYNAMO II User's Manual, MIT Press, 1973

D.L.Meadows, Dynamics of commodity production cycles, Cambridge, Mass. Wright Allen Press, 1970

C.A.Danten & L.M.Valentine, Business cycles and forecasting, South-Western Publishing Co., 1974

J.T.Schwartz, Lectures on the mathematical methods in analytical economics, Gordon & Breach, 1961

J.T.Schwartz, Theory of money, Gordon & Breach, 1965

10.6. Aufgaben

1. Man simuliere das DYNAMO-Modell von Kapitel 10.3. statt mit der linearen Input-Output Beziehung gemäß Figur 10-1, mit der nichtlinearen Beziehung gemäß Figur 10-2.
2. Man überlege sich andere antizyklische Politiken beim exogenen Konsum bzw. den Löhnen und untersuche deren Einfluß auf den Konjunkturverlauf.

11. PERSONALPROGNOSEMODELL IN EINEM KRANKENHAUS

11.1. Einleitung

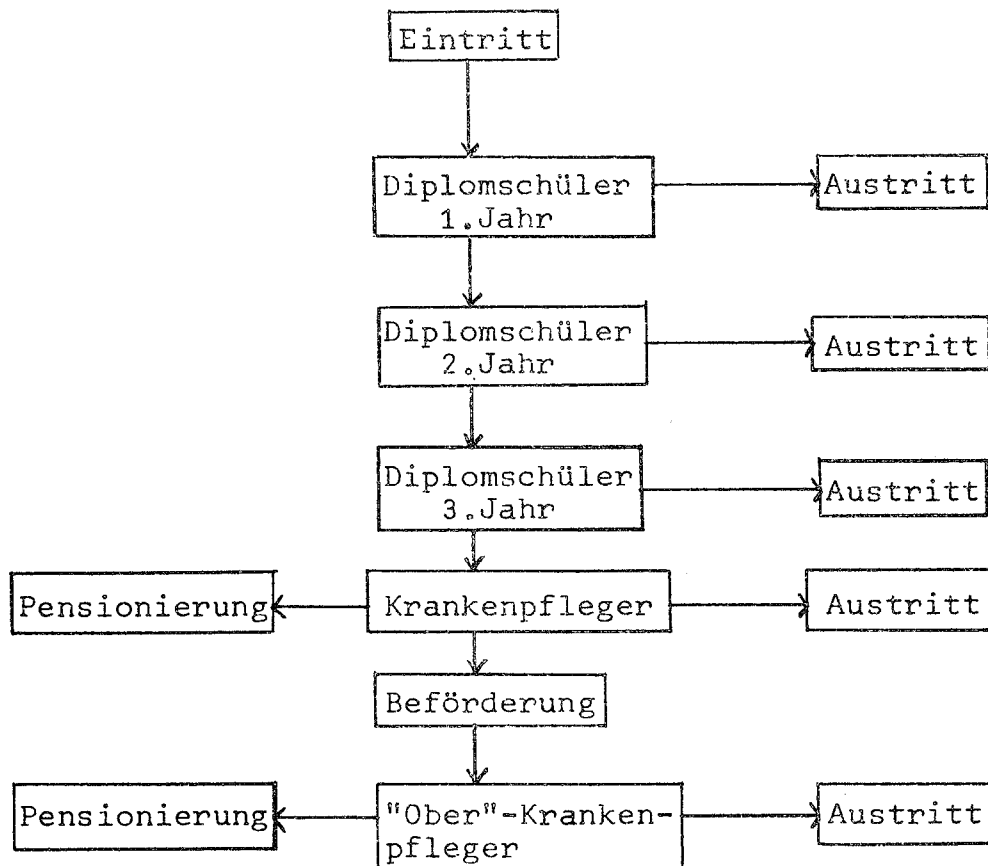
Zum Abschluß soll noch über eine konkrete Fallstudie berichtet werden, welche im Zusammenhang mit einer organisationssoziologischen Studie an einem psychiatrischen Spital auftrat. Es ergab sich das Problem, den Personalstand an qualifiziertem Pflegepersonal für die nächsten 10-20 Jahre zu prognostizieren, da der Verdacht bestand, daß in diesem Zeitraum dieser Personalstock sich drastisch reduzieren würde. Dies genauer zu be- oder widerlegen war Aufgabe des Simulationsmodells. Im nächsten Kapitel soll das Modell verbal genau beschrieben werden, die Datenlage erörtert und damit die Möglichkeiten und Grenzen des Modells diskutiert werden. Daran anschließend entwickeln wir das quantitative Modell inklusive dem Computer-Programm.

11.2. Beschreibung des Ist-Zustandes

Der interessierende Personalbereich ist der des Pflegepersonals. Dieses wird wiederum unterteilt in diplomiertes Pflegepersonal und Hilfspersonal. Ersteres weist eine 3-jährige Ausbildung, letzteres nur eine kurze Einschulung auf. Das Interesse konzentriert sich nun aber auf das diplomierte Personal, da nur dieses über die entsprechende Qualifikation verfügt und gerade hier ein starker Personalschwund in den nächsten Jahren vermutet wird. Betrachten wir den Karriereverlauf des diplomierten Personals näher. Dieses rekrutiert sich ausschließlich aus der spitaleigenen Krankenpflegerschule, welche in einem 3-jährigen Lehrgang zum diplomierten Krankenpfleger ausbildet.

Rekrutierung von anderen als der spitalseigenen Schule oder von anderen Spitälern erfolgt nicht, aus Gründen, die nicht Gegenstand der Untersuchung sind.

Innerhalb der Berufsgruppe der diplomierten Krankenpfleger gibt es genau zwei Berufspositionen. Die untere Position ist diejenige, in welche jeder Diplomierte automatisch gelangt. In die obere Position kann man nur gelangen, falls dort ein Posten frei wird - ist also kein automatischer Aufstieg auf Grund von Dienstalter u.ä. In Figur 11-1 ist diese Berufskarriere nochmals graphisch dargestellt. Eine wichtige Aufgabe, das ergibt sich aus Figur 11-1, muß es nun sein, die Austrittswahrscheinlichkeit bzw. Pensionierungsrate zu ermitteln, dann läßt sich, ausgehend von gegebenen Eintrittten pro Jahr, der Personalstand in den kommenden Jahren ermitteln. An Datenmaterial stehen nun die Personalakte aller diplomierten Pfleger der letzten 10 Jahre zur Verfügung, aus denen neben Eintrittsdatum und gegebenenfalls Austrittsdatum auch Geschlecht, Alter, Zivilstand, Staatsbürgerschaft und Berufsposition hervorgeht.



Figur 11-1
Krankenpfleger-Karriere

Auf Grund dieser Daten können nun die Austrittswahrscheinlichkeiten in den verschiedenen Karrierestufen geschätzt werden. Die Pensionierung hingegen erfolgt zwangsläufig bei Erreichen der entsprechenden Altersstufe. Nun kann aber auch ein Einfluß nicht nur der Karrierestufen auf die Austrittswahrscheinlichkeit statistisch erhärtet werden, sondern auch Alter, Dienstalter, Geschlecht, Zivilstand evtl. auch die Kategorie In- bzw. Ausländer dürfte den Entscheid, das Spital zu verlassen, beeinflussen.

Indes sind nicht genügend Daten vorhanden, um all diese Einflüsse klar herauszuarbeiten, weswegen man sich beschränkt auf die folgenden Kategorien zur Festlegung der Austrittswahrscheinlichkeit pro Jahr:

- Berufsposition
- Geschlecht
- Alter

Dienstalter wurde weggelassen (im wesentlichen deckt es sich mit der Kategorie Alter) ebenso Nationalität, da nur ganz wenige Ausländer überhaupt diplomierte Krankenpfleger sind, aber auch Zivilstand, da zuwenig Information über dessen Änderung mit der Zeit bekannt ist. Die Schätzung der Austrittswahrscheinlichkeiten könnten wesentlich vereinfacht werden, könnte man die Unabhängigkeit der "Ereignisse" der einzelnen Kategorien voneinander postulieren, denn dann gilt bekanntlich

$$p(a,b) = p(a).p(b) .$$

Gehen wir davon aus, daß es 5 Berufskategorien (siehe Figur 11-1) und 10 Altersstufen (15-65 Jahre in 5-Jahres-Stufen) gibt, so müßte man bei Unabhängigkeit

$$10 + 5 + 2 = 17$$

Wahrscheinlichkeiten, in unserem Falle aber

$$10.5.2 = 100 \quad (11-1)$$

Wahrscheinlichkeiten berechnen.

Eine wichtige Annahme, wenn nicht die Hauptannahme überhaupt bei Prognosemodellen ist das Prinzip, daß die aus den Daten der Vergangenheit abgeleiteten Trends sich nicht in der Zukunft ändern, andernfalls das Prognosemodell keinerlei Prognosefähigkeit mehr besitzt. Prognosen sollten daher stets mit dem Satz beginnen: "Unter der Annahme, daß die Gesetze und Trends der Vergangenheit auch in Zukunft gültig sind ..."

Um das Modell nicht noch mehr zu komplizieren bzw. um nicht mehr an Information aus den vorhandenen Daten herauszubekommen, als eigentlich drinnen ist, müssen wir ein konstantes Verhalten des Pflegepersonals in den letzten 10 Jahren (dem Datenerhebungszeitraum) und dem Prognosezeitraum a priori festlegen.

Zur Schätzung der 100 Wahrscheinlichkeiten gemäß (11-1) müssen alle Daten der letzten 10 Jahre als gleich angesehen werden. Sollte auch noch ein Trend geschätzt werden, also die Abhängigkeit der Wahrscheinlichkeiten von der Zeit, so reichen die vorhandenen Daten nicht aus. Unser verbales Modell ist nun soweit gediehen, daß wir es zu einem Simulationsmodell aufbereiten können.

11.3. Das Simulationsmodell

Da die "Logik" des Modells nicht aufwendig ist, ergibt sich das Simulationsprogramm auf einfache Art. Jeder Krankenpfleger wird beschrieben durch Angabe seines Alters, Geschlecht und Berufsposition. Für diesen so definierten Krankenpfleger kennen wir seine Austrittswahrscheinlichkeit pro Jahr (wir wählen als Zeiteinheit ein Jahr), gegebenenfalls Pensionierung. Für die Besetzung von frei werdenden Posten der oberen Berufsposition soll jeder Krankenpfleger der unteren Berufsposition gleiche Chancen haben - es wird also zufällig einer ausgewählt. Jedes Jahr tritt eine vorgegebene Anzahl Personen in die 1.Klasse des 3-jährigen Lehrgangs ein, mit gegebenen Parametern (Alter und Geschlecht). Nun wird für jede Person gemäß ihrer Charakteristiken ihr neuer Zustand in einem Jahr bestimmt (Austritt, Pensionierung, Beförderung oder status quo). Dies wird für den erwünschten Prognosezeitraum (ca. 20 Jahre) dementsprechend oft wiederholt. Da es sich aber um einen instationären, stochastischen Prozeß handelt, müssen diese 20 Jahre so oft simuliert werden, bis die geschätzten Erwartungswerte von Anzahl Krankenpfleger im Jahr X mit genügender Genauigkeit bestimmt werden können. Um die

Validität des Modells zu testen und eventuell Korrekturen vorzunehmen wird so vorgegangen: Zum Schätzen der Parameter werden die Daten 1965-1973 herangezogen. Dem Modell wird nun der Personalstand, wie er 1973 war eingegeben und überprüft, wie gut die Prognose für 1974-75 im Vergleich zu dem bekannten tatsächlichen Personalstand ist. Bei gegebener Übereinstimmung ist dies ein Hinweis auf die Realitätstreue des Modells. Da weder SIMSCRIPT noch DYNAMO als Programmiersprache besonders geeignet für dieses Modell erscheinen, wird FORTRAN gewählt. Der Input in das Programm sind die geschätzten Wahrscheinlichkeiten und der Anfangszustand des Systems, also die Krankenpfleger mit deren Charakteristiken zu einem bestimmten Zeitpunkt. Eine spezielle Subroutine soll die jährlichen Kursanfänger "erzeugen". Als Output soll das Modell für eine definierte Anzahl Jahre die Anzahl Krankenpfleger in jedem Jahr (Erwartungswert und Varianz) evtl. mit Altersstruktur schätzen. Um die Reaktion des Systems auf verschiedene Personalpolitiken testen zu können, soll insbesondere möglich sein

- Variierung der Anzahl Eintretender pro Jahr
- Variierung der Anzahl Posten in der oberen Berufsposition
- Variierung einzelner Austrittsraten.

Das Computerprogramm lautet

```
C      WICHTIGE GROESSEN
C      KTART(I,J)..PFLEGEPERSONAL ZU BEGINN DES SIMULATIONS=
C      ZEITRAUMES, WOBEI I DER FORTLAUFENDEN
C      NUMERIERUNG DIENT.
C      KTART(I,1)=1..SCHUELER IM 1. JAHR
C      =2..SCHUELER IM 2. JAHR
C      =3..SCHUELER IM 3. JAHR
C      =4..NORMALER PFLEGER
C      =5..OBERPFLEGER
C      KTART(I,2)=1..MAENNLICH
C      =2..WEIBLICH
C      KTART(I,3)=ALTER
C      P(I,J,K)..AUSTRITTSWAHRSCHEINLICHKEIT DES PFLEGERS X
C      INNERT EINES JAHRES, FALLS PFLEGER X DEN BERUFSSTATUS I,
C      DAS GESCHLECHT J UND DAS ALTER K AUFWEIST
C      ZUF(I,J)..WAHRSCHEINLICHKEIT, DASS JEMAND MIT GESCHLECHT I
C      UND ALTER J+17 ALS PFLEGESCHUELER EINTRITT (SUMME UEBER
C      ALLE WAHRSCHEINLICHKEITEN IST 1)
C      E(I)..GESCHAEZTER ERWARTUNGSWERT DER ANZAHL PFLEGER IM
C      JAHRE I
C      S(I)..GESCHAEZTE STANDARDABWEICHUNG VON E(I)
C
C      DIESES HAUPTPROGRAMM DIENT NUR DEM AUFRUF VON SUBROUTINEN
C
C      DIMENSION KTART(1000,3),P(5,2,10),ZUF(2,20),E(40),S(40)
C      CALL LESEN(KTART,P,ZUF,N,JAHR,EPS)
C      CALL SCHAEZ(KTART,P,ZUF,N,JAHR,E,S,EPS)
C      CALL DRUCK(E,S,JAHR)
C      END
```

SUBROUTINE LESEN(KTART,P,ZUF,N,JAHR,EPS)

SUBROUTINE ZUM EINLESEN DER DATEN

WICHTIGE GROESSEN

N..ANZAHL PFLEGER

JAHR..ANZAHL ZU SIMULIERENDER JAHRE

EPS..ABBRUCHKRITERIUM - BESCHRAENKT DIE STANDARDAB-
WEICHUNG VON E(I)

DIMENSION KTART(1000,3),P(5,2,10),ZUF(2,20)

READ 5,N,JAHR,EPS

PRINT 5,N,JAHR,EPS

5 FORMAT(2I3,F4.2)

READ 10,((KTART(I,J),J=1,3),I=1,N)

READ 15,(((P(I,J,M),I=1,5),J=1,2),M=1,10)

READ 15,((ZUF(I,J),I=1,2),J=1,20)

PRINT 10,((KTART(I,J),J=1,3),I=1,N)

PRINT 15,(((P(I,J,M),I=1,5),J=1,2),M=1,10)

PRINT 15,((ZUF(I,J),I=1,2),J=1,20)

10 FORMAT(39I2)

15 FORMAT(20F4.2)

RETURN

END

SUBROUTINE SCHAEZ(KTART,P,ZUF,N,JAHR,E,S,EPS)

SUBROUTINE ZUM SCHAETZEN VON E(I) UND S(I) MIT
ABBRUCHKRITERIUM (SIEHE LABEL 20)

WICHTIGE GROESSEN

L(I)..ANZAHL PFLEGER IM JAHRE I DES SIMULATIONSZEITRAUMES

KX..LETZTE ERZEUGTE ZUFALLSZAHL (0<KX<2**35)

DIMENSION KTART(1000,3),P(5,2,10),ZUF(2,20),E(40),S(40)

DIMENSION L(40),E1(40),S1(40)

KX=0

DO 5 J=1,JAHR

E1(J)=0

5 S1(J)=0

I=0

10 I=I+1

CALL SIM(KTART,P,ZUF,KX,N,JAHR,L)

DO 15 J=1,JAHR

E1(J)=E1(J)+L(J)

S1(J)=S1(J)+L(J)*L(J)

E(J)=E1(J)/I

15 S(J)=SQRT(S1(J)/I-E(J)*E(J))

IF(I.LT.20) GOTO 10

DO 20 J=1,JAHR

```
20 IF(S(J)/SQRT(FLOAT(I)).GT.EPS*E(J)) GOTO 10
RETURN
END
```

```
SUBROUTINE DRUCK(E,S,JAHR)
```

```
C
C SUBROUTINE ZUM DRUCKEN DER RESULTATE
C
```

```
    DIMENSION E(1),S(1)
    DO 5 J=1,JAHR
5    PRINT 10,J,E(J),S(J)
10   FORMAT(' ANZAHL PFLEGER IM',I3,'-TEN JAHR =',F5.0,' BEI EINER
XSTREUUNG VON',F8.3)
    RETURN
END
```

```
SUBROUTINE SIM(KTART,P,ZUF,KX,N,JAHR,L)
```

```
C
C SUBROUTINE ZUR SIMULATION DER JAEHRLICHEN VER=
C AENDERUNG DES PFLEGERBESTANDES
C
C WICHTIGE GROESSEN
C PFLEG(I,J)..AKTUELLER PFLEGERBESTAND (GENAUE BE=
C SCHREIBUNG ANALOG ZU KTART(I,J) )
C SCHUEL(I,J)..EINTRETENDE PFLEGESCHUELER, MIT FORT=
C LAUFENDER NUMERIERUNG I UND TYPISIERUNG J ANALOG
C ZU KTART(I,J)
C KY..ANZAHL OFFENER OBERPFLEGER-STELLEN
C NRAND(..)..ZUFALLSZAHLENGENERATOR MIT VORHERGEHENDER
C ZUFALLSZAHLS ALS INPUT
C M..ANZAHL EINTRETENDER PFLEGERSCHUELER (29<M<71)
    DIMENSION KTART(1000,3),P(5,2,10),ZUF(2,20),PFLEG(1000,3)
    DIMENSION SCHUEL(100,3),L(40)
    DO 5 J=1,N
    DO 10 J1=1,3
    PFLEG(J,J1)=KTART(J,J1)
10   CONTINUE
5    CONTINUE
    L(1)=N
    KY=0
    DO 15 J=1,JAHR
    KX=NRAND(KX)
    Z=KX/2.**35.
    M=30+INT(Z*40)
    CALL STAT(PFLEG,P,N,KX,KY)
    CALL EIN(SCHUEL,M,ZUF,KX)
    CALL AUS(PFLEG,SCHUEL,N,M)
    J1=J+1
15   L(J1)=N
    N=L(1)
    RETURN
```


END

SUBROUTINE STAT(PFLEG,P,N,KX,KY)

SUBROUTINE ZUM BESTIMMEN DES NEUEN STATUS DER PFLEGER

ALLE WICHTIGEN GROESSEN SIND IN VORANGEHENDEN
SUBROUTINEN SPEZIFIZIERT

DIMENSION PFLEG(1000,3),P(5,2,10)

DO 5 J=1,N

PFLEG(J,3)=PFLEG(J,3)+1

IF(PFLEG(J,1).LT.4) PFLEG(J,1)=PFLEG(J,1)+1

IF(PFLEG(J,3).GE.63) GOTO 10

IF(PFLEG(J,3).GE.58.AND.PFLEG(J,2).GE.2) GOTO 10

KX=NRAND(KX)

Z=KX/2.**35.

I1=PFLEG(J,1)

I2=PFLEG(J,2)

I3=INT((PFLEG(J,3)-18.)/5.)+1

IF(Z.GT.P(I1,I2,I3)) GOTO 5

10 PFLEG(J,2)=0

IF(PFLEG(J,1).GE.5) KY=KY+1

5 CONTINUE

25 IF(KY.EQ.0) GOTO 15

KX=NRAND(KX)

Z=KX/2.**35.

I1=1+INT(Z*(N-0.0001))

IF(PFLEG(I1,1).EQ.4.AND.PFLEG(I1,2).GT.0) GOTO 20

GOTO 25

20 PFLEG(I1,1)=5

KY=KY-1

GOTO 25

15 CONTINUE

RETURN

END

SUBROUTINE EIN(SCHUEL,M,ZUF,KX)

SUBROUTINE ZUM "ERZEUGEN" DER PFLEGESCHUELER

DIMENSION SCHUEL(100,3),ZUF(2,20)

DO 5 J=1,M

SCHUEL(J,1)=1

KX=NRAND(KX)

Z=KX/2.**35.

DO 10 J1=1,2

DO 15 J2=1,20

IF(Z.GT.ZUF(J1,J2)) GOTO 15

SCHUEL(J,2)=J1

SCHUEL(J,3)=J2+17

GOTO 5

```
15  CONTINUE
10  CONTINUE
5   CONTINUE
    RETURN
    END
```

SUBROUTINE AUS(PFLEG, SCHUEL, N, M)

```
C
C  SUBROUTINE ZUR EINGLIEDERUNG DER SCHUELER SCHUEL(I, J)
C  IN DEN PFLEGERBESTAND PFLEG(K, L) UND FESTSTELLUNG DES
C  AKTUELLEN PFLEGERBESTANDES N
C
```

```
    DIMENSION PFLEG(1000, 3), SCHUEL(100, 3)
    I=0
    J=0
20  I=I+1
    IF(I.GT.M) GOTO 5
15  J=J+1
    IF(J.GT.N) GOTO 10
    IF(PFLEG(J, 2).GT.0) GOTO 15
    PFLEG(J, 1)=SCHUEL(I, 1)
    PFLEG(J, 2)=SCHUEL(I, 2)
    PFLEG(J, 3)=SCHUEL(I, 3)
    GOTO 20
10  N=N+1
    PFLEG(N, 1)=SCHUEL(I, 1)
    PFLEG(N, 2)=SCHUEL(I, 2)
    PFLEG(N, 3)=SCHUEL(I, 3)
    I=I+1
    IF(I.GT.M) GOTO 25
    GOTO 10
5   J=J+1
    IF(J.GE.N) GOTO 25
    IF(PFLEG(J, 2).GT.0) GOTO 5
    PFLEG(J, 1)=PFLEG(N, 1)
    PFLEG(J, 2)=PFLEG(N, 2)
    PFLEG(J, 3)=PFLEG(N, 3)
    N=N-1
    GOTO 5
25  CONTINUE
    RETURN
    END
```

Da das Resultat mit den wirklichen Daten nicht wiedergegeben werden kann, soll nur ein Prognosebeispiel für einen Zeitraum von 35 Jahren hier dargestellt werden.

ANZAHL	PFLEGER	IM	1-TEN	JAHR	=	650.	BEI	EINER	STREUUNG	VON	.000
ANZAHL	PFLEGER	IM	2-TEN	JAHR	=	633.	BEI	EINER	STREUUNG	VON	10.497
ANZAHL	PFLEGER	IM	3-TEN	JAHR	=	619.	BEI	EINER	STREUUNG	VON	13.382
ANZAHL	PFLEGER	IM	4-TEN	JAHR	=	580.	BEI	EINER	STREUUNG	VON	16.995
ANZAHL	PFLEGER	IM	5-TEN	JAHR	=	501.	BEI	EINER	STREUUNG	VON	19.696
ANZAHL	PFLEGER	IM	6-TEN	JAHR	=	485.	BEI	EINER	STREUUNG	VON	21.599
ANZAHL	PFLEGER	IM	7-TEN	JAHR	=	464.	BEI	EINER	STREUUNG	VON	22.225
ANZAHL	PFLEGER	IM	8-TEN	JAHR	=	465.	BEI	EINER	STREUUNG	VON	26.815
ANZAHL	PFLEGER	IM	9-TEN	JAHR	=	467.	BEI	EINER	STREUUNG	VON	24.898
ANZAHL	PFLEGER	IM	10-TEN	JAHR	=	474.	BEI	EINER	STREUUNG	VON	26.315
ANZAHL	PFLEGER	IM	11-TEN	JAHR	=	463.	BEI	EINER	STREUUNG	VON	28.874
ANZAHL	PFLEGER	IM	12-TEN	JAHR	=	467.	BEI	EINER	STREUUNG	VON	34.548
ANZAHL	PFLEGER	IM	13-TEN	JAHR	=	477.	BEI	EINER	STREUUNG	VON	31.294
ANZAHL	PFLEGER	IM	14-TEN	JAHR	=	451.	BEI	EINER	STREUUNG	VON	26.697
ANZAHL	PFLEGER	IM	15-TEN	JAHR	=	453.	BEI	EINER	STREUUNG	VON	25.240
ANZAHL	PFLEGER	IM	16-TEN	JAHR	=	459.	BEI	EINER	STREUUNG	VON	20.865
ANZAHL	PFLEGER	IM	17-TEN	JAHR	=	467.	BEI	EINER	STREUUNG	VON	25.297
ANZAHL	PFLEGER	IM	18-TEN	JAHR	=	476.	BEI	EINER	STREUUNG	VON	21.140
ANZAHL	PFLEGER	IM	19-TEN	JAHR	=	481.	BEI	EINER	STREUUNG	VON	25.947
ANZAHL	PFLEGER	IM	20-TEN	JAHR	=	482.	BEI	EINER	STREUUNG	VON	24.777
ANZAHL	PFLEGER	IM	21-TEN	JAHR	=	491.	BEI	EINER	STREUUNG	VON	26.411
ANZAHL	PFLEGER	IM	22-TEN	JAHR	=	504.	BEI	EINER	STREUUNG	VON	25.785
ANZAHL	PFLEGER	IM	23-TEN	JAHR	=	510.	BEI	EINER	STREUUNG	VON	29.255
ANZAHL	PFLEGER	IM	24-TEN	JAHR	=	489.	BEI	EINER	STREUUNG	VON	35.483
ANZAHL	PFLEGER	IM	25-TEN	JAHR	=	489.	BEI	EINER	STREUUNG	VON	35.385
ANZAHL	PFLEGER	IM	26-TEN	JAHR	=	493.	BEI	EINER	STREUUNG	VON	36.487
ANZAHL	PFLEGER	IM	27-TEN	JAHR	=	498.	BEI	EINER	STREUUNG	VON	34.951
ANZAHL	PFLEGER	IM	28-TEN	JAHR	=	499.	BEI	EINER	STREUUNG	VON	33.918
ANZAHL	PFLEGER	IM	29-TEN	JAHR	=	508.	BEI	EINER	STREUUNG	VON	35.064
ANZAHL	PFLEGER	IM	30-TEN	JAHR	=	513.	BEI	EINER	STREUUNG	VON	34.860
ANZAHL	PFLEGER	IM	31-TEN	JAHR	=	515.	BEI	EINER	STREUUNG	VON	33.814
ANZAHL	PFLEGER	IM	32-TEN	JAHR	=	517.	BEI	EINER	STREUUNG	VON	30.804
ANZAHL	PFLEGER	IM	33-TEN	JAHR	=	518.	BEI	EINER	STREUUNG	VON	28.911
ANZAHL	PFLEGER	IM	34-TEN	JAHR	=	507.	BEI	EINER	STREUUNG	VON	28.402
ANZAHL	PFLEGER	IM	35-TEN	JAHR	=	498.	BEI	EINER	STREUUNG	VON	28.921

11.4. Literatur

G.H.Orcutt, M.Greenberger, J.Kobel & A.M.Rivlin, Microanalysis of Socioeconomic Systems: A Simulation Study, Harper & Row, New York, 1965.